Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет Інфокомунікацій
(повна назва)
Кафедра Інфокомунікаційної інженерії імені В.В. Поповського
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

рівень вищої освіти другий (магістр)

Дослідження кібербезпеки технології Інтернету речей (IoT) з використанням хмарних обчислень (Investigating cyber security of Internet of Things (IoT) technology using cloud computing)
(тема)

Виконав:
студент 5 курсу, групи АМСЗІімв-20-1
Ларабі Фоуад
(прізвище, ініціали)
Спеціальність: 125 Кібербезпека
(код і повна назва спеціальності)
Освітня програма: Адміністративний менеджмент у сфері захисту інформації
(повна назва освітньої програми)
Керівник: професор кафедри ІКІ ім. В.В. Поповського
Марчук В.С.
(посада, прізвище, ініціали)

Допускається до захисту
Зав. кафедри _____ Лемешко О.В.
(підпис) (прізвище, ініціали)

2023 р.

Кваліфікаційна робота не містить відомостей, що заборонені до відкритого друку

Студент 5 курсу

групи АМСЗІімв-20-1  Ларабі Фоуад

(підпис) (ініціали, прізвище)

Керівник  В.С. Марчук

(підпис) (ініціали, прізвище)

Харківський національний університет радіоелектроніки

Факультет Інфокомунікацій
(повна назва)
Кафедра Інфокомунікаційної інженерії імені В.В. Поповського
(повна назва)
Рівень вищої освіти другий (магістр)
Спеціальність 125 Кібербезпека
(код і повна назва)
Освітня програма Адміністративний менеджмент у сфері захисту інформації
(повна назва)

ЗАТВЕРДЖУЮ
Зав. кафедри_____
(підпис)
«____» _____2023 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

студенту Ларабі Фоуад
(прізвище, ім'я, по батькові)

1. Тема роботи: Дослідження кібербезпеки технології Інтернету речей (IoT) з використанням хмарних обчислень (Investigating cyber security of Internet of Things (IoT) technology using cloud computing)
затверджена наказом по університету від «_31__» __10__ 2022 р. №_1434_ Ст.
2. Термін подання студентом роботи до екзаменаційної комісії __22.01._2023 р.__
3. Вихідні дані до роботи: Вимоги стандарту ISO / IEC 27001 A.10.10.2 , щодо виявлення атак. Допустимість використання програмного та апаратного забезпечення (стандарт ISO / IEC 27001 A.12.4.1), контроль змін в інформаційній системі (ISO / IEC 27001 A.12.5.1),), а також основні протоколи, що забезпечують функціонування мереж IoT з використанням хмарних технологій.
4. Перелік питань, що потрібно опрацювати в роботі:
1) Огляд сучасного стану технологій та протоколів, що забезпечують функціонування мереж IoT з використанням хмарних технологій.
2) Аналіз вразливостей мереж IoT з використанням хмарних технологій.
3) Дослідження багатоступеневого захисту мереж IoT з інтеграцією технології Fog Computing та Cloud computing.

5. Перелік графічного матеріалу із зазначенням креслень, плакатів, комп'ютерних ілюстрацій: Демонстраційний матеріал у вигляді ppt-презентації

6. Консультанти розділів роботи

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		(підпис)	(дата)
Основна частина	професор Марчук Володимир Степанович		22.01.23

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	Отримання завдання	31.10.22	Виконано
2	Збір матеріалів для дослідження	15.11.22	Виконано
3	Розробка 1 розділу	20.12.22	Виконано
4	Розробка 2 розділу	01.01.23	Виконано
5	Розробка 3 розділу	10.01.23	Виконано
6	Розробка 4 розділу	15.01.23	Виконано
7	Оформлення атестаційної роботи	22.01.23	Виконано

Дата видачі завдання 31 жовтня 2022 року

Студент  Ларабі Фоуад
(підпис) (прізвище, ініціали)

Керівник роботи  професор Марчук В.С.
(підпис) (посада, прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка: 66 с., 16 рис., 3 табл., 11 джерел.

ТЕХНОЛОГІЯ IoT, ХАРАКТЕРИСТИКИ І АРХІТЕКТУРА СИСТЕМ IoT, МЕРЕЖІ FOG COMPUTING І CLOUD COMPUTING, ПРОТОКОЛИ МЕРЕЖ IoT, ВРАЗЛИВОСТІ МЕРЕЖ IoT, БАГАТОСТУПЕНЕВИЙ ЗАХИСТ, ЙМОВІРНІСТЬ ЗАГРОЗ.

Об’єкт дослідження – процеси функціонування телекомунікаційних систем IoT з використанням хмарних обчислень.

Предмет дослідження – характеристиики телекомунікаційних систем IoT з використанням хмарних обчислень.

Мета роботи – аналіз архітектури і параметрів технологій телекомунікаційних систем IoT з використанням хмарних обчислень, а також вдосконалення методів інформаційного захисту таких систем.

Методи дослідження – емпіричний аналіз, порівняння, аналітичне моделювання.

На сьогоднішній день тема впровадження нового покоління мереж IoT з використанням хмарних обчислень є актуальною, оскільки в наш час відбувається суттеве збільшення кількості користувачів і потоків інформації в мережах IoT.

У роботі виконано аналіз сучасних мереж IoT з використанням технологій Fog Computing і Cloud Computing. Запропоновано вдосконалення методу інформаційного захисту систем IoT. Виконано розрахунок ймовірності відбиття загроз в системі IoT. Розроблені рекомендації по збільшенню захисту систем IoT.

ABSTRACT

Explanatory note: 66 p., 16 fig., 3 table, 11 sources.

IoT TECHNOLOGY, CHARACTERISTICS AND ARCHITECTURE OF IoT SYSTEMS, FOG COMPUTING AND CLOUD COMPUTING NETWORKS, IoT NETWORK PROTOCOLS, IoT NETWORK VULNERABILITIES, MULTI-LEVEL PROTECTION, PROBABILITY OF THREATS.

The object of research is the processes of functioning of IoT telecommunication systems using cloud computing.

The subject of the research is the characteristics of IoT telecommunication systems using cloud computing.

The purpose of the work is to analyze the architecture and technology parameters of IoT telecommunication systems using cloud computing, as well as to improve the methods of information protection of such systems.

Research methods – empirical analysis, comparison, analytical modeling.

Today, the topic of introducing a new generation of IoT networks using cloud computing is relevant, as nowadays there is a substantial increase in the number of users and information flows in IoT networks.

The paper analyzes modern IoT networks using Fog Computing and Cloud Computing technologies. It is proposed to improve the method of information protection of IoT systems. The probability of threat reflection in the IoT system has been calculated. Recommendations for increasing the protection of IoT systems have been developed.

CONTENTS

LIST OF ABBREVIATIONS, SYMBOLS, UNITS AND TERMS

ABS - Alert Based Systems
ARP - Address Resolution Protocol
AES - Advanced Encryption Standard
API – application programming interface
CoAP – constrained application protocol
CNI - Critical National Infrastructure
DDoS – distributed denial of service
DDS – data distribution service
DoS - Denial of Service
DPMU – distinct program management unit
DTLS – datagram transport layer security
ECC – elliptic curve cryptography
HTTPS – hypertext transfer protocol secure
IaaS - Infrastructure as a service
IDS - Intrusion Detection System
IoT – internet of things
IP – internet protocol
IPS - Intrusion Prevention System
M2M – machine to machine
MITM - Man-In-The-Middle
MQTT – message queue telemetry transport
OMG – object management group.
OTA – over-the-air
PaaS – platformas as service
QoS – quality of service
RFID – radio frequency identification
RSA – rivest-shamir-adleman
SaaS – software as a service
SASL – simple authentication and security layer
SSL – secure sockets layer
TCP – transmission control protocol

TLS – transport layer security
UDP – user datagram protocol
UUID – universally unique identifiers
VPN – virtual private network
WLAN – wireless local area network
WSN – wireless sensor networks
XML – extensible markup language
XMPP – extensible messaging and presence protocol

INTRODUCTION

The Internet of Things (IoT) is defined as the system of interconnected electronic devices embedded with software, sensors, actuators, and network connectivity which enable them to connect and exchange data. Smart devices such as wearable devices, home appliances, alarms and camera systems routinely collect personal information and provide various functionalities which automate and support our daily activities and needs.

As a result, the popularity of such devices has significantly increased over the past few years. This is due to their affordability, as well as their ubiquitous connectivity which allows them to communicate and exchange information with other technologies, their intelligence, and their decision-making capabilities to invoke actions. This provides seamless user experiences which significantly enhances people's every day lives and is demonstrated by how prominent such devices are today.

The proliferation of smart devices is not only within the domestic environment, but it is also the driving force behind the development of an interconnected knowledge based world; our economies, societies, machinery of government, and Critical National Infrastructure (CNI).

However, although these concepts support everyday life tasks, their dependency on Information Communication Technology (ICT) and IoT devices introduce severe security risks. Whereas security protocols and best practices for traditional Information Technology (IT) is well-understood and broadly adopted,security for IoT devices is nascent and is rarely sufficient.

Cyber attacks against IoT can lead to disastrous effects including personal information leakage, damage to hardware, disrupting the system's availability, causing system blackouts, and even physically harm individuals. Thus, the scale of the impact of the attacks performed on IoT networks can vary significantly depending on the targeted device.

Subsequently, given that IoT devices have a direct impact on our lives, security and privacy considerations must become a high priority .

Two of the main reasons that make IoT devices vulnerable to cyber attacks include their limitations in computational power and their heterogeneity. More specifically, it is generally not feasible for IoT devices with restricted computational

power, memory, radio bandwidth, and battery resource to execute computationally intensive and latency sensitive security tasks that generate heavy computation and communication load.

As a result, it is not possible to employ complex and robust security measures. In addition, the heterogeneity which surrounds IoT devices in terms of hardware, software and protocols poses a great challenge towards developing and deploying security mechanisms that can endure with the scale and range of devices. Consequently, it is evident that there is a major gap between security.

This qualification work provides an overview of networks with IoT technology and the use of cloud services. The architecture of such networks and features of their functioning are given. An analysis of the vulnerabilities of networks with IoT technology and the use of cloud services was carried out. Methods for dealing with vulnerabilities are proposed and calculations of the probabilities of repelling threats are carried out.

1 OVERVIEW OF THE CURRENT STATE OF TECHNOLOGIES ENSURING THE FUNCTIONING OF IoT NETWORKS USING CLOUD TECHNOLOGIES

1.1 General provisions

The term IoT was first introduced as an idea by Kevin Ashton in 1999. Today, IoT technology is used in many industries and in millions of devices and systems.

IoT is a technology that implements the process of interaction between intelligent devices and devices connected to the Internet. All of them create an iot network. IoT devices typically interact not with humans, but with other devices and software.

The size, complexity and cyber security of these devices and systems largely depends on the number of interactions they have with other devices in the system. Now these devices are used both at home and in offices and industry.

Industries that use IoT include:

- consumer services (smartphones, smart watches, smart homes);
- business applications (smart security cameras, trackers for transport);
- government programs (road traffic monitoring, natural disaster warnings).

The IoT technology uses various existing network technologies. One of these is the RFID (Radio Frequency Identification) technology, which is based on the use of the radio range. This technology allows the use of microchips that allow data exchange in wireless networks. Microchips use tags, or in other words labels, as an identifier that allows automatic identification of an object. A tag can be both passive and active.

Passive RFID tags do not consume energy and therefore do not need their own battery, mostly they are used as a simple identifier and transfer all the work to the reader of such RFID tags.

Active RFID tags are equipped with their own battery, which allows them to communicate with other devices themselves.

Another technology used in IoT is WSN (Wireless Sensor Networks). This system consists of cheap, low-power, autonomous and small devices that are geographically distributed in different parts of the area. They use sensors to monitor the environment. The WSN system also includes a gateway that provides wireless communication with distributed network nodes and the main network.

These technologies allow you to store data accumulated through the Internet of Things. The data is stored on a decentralized server that uses computer resources that can be accessed when needed. In addition, cloud technologies are used to calculate large data packages.

IoT is a system that includes several components that are interconnected by systems that perform data processing tasks in real time. Iot consists of three layers such as [1]:

- applied level;
- level of communication;
- physical level.

A schematic representation of the layers and protocol stacks corresponding to each layer can be seen in Figure 1.1 [1].

As can be seen from fig. 1.1, the physical layer is the lowest layer of the IoT layer stack. It is responsible for the connection between physical things or, in other words, between Internet of Things devices and their identification. The main requirement for such devices is to equip them with communication technologies that ensure their connection to each other directly using the Internet. In addition, each device must have a unique identifier or tag that will ensure its connection to the network. Usually, such identifiers are arranged in the system in the form of a chip, the main purpose of which is to identify the device itself.

The next level is the level of communication. As in most layers of other models, it includes network interfaces, communication channels and network management. The main goal of this level is to ensure communication between all devices in the iot system using Internet protocols. The main protocols used at this level are MQTT (Message Queue Telemetry Transport) and COAP (Constrained Application Protocol). Also at this level, the data collected from the physical layer is transformed into data that can be processed and transmitted over the wired Internet network or through the mobile Internet.

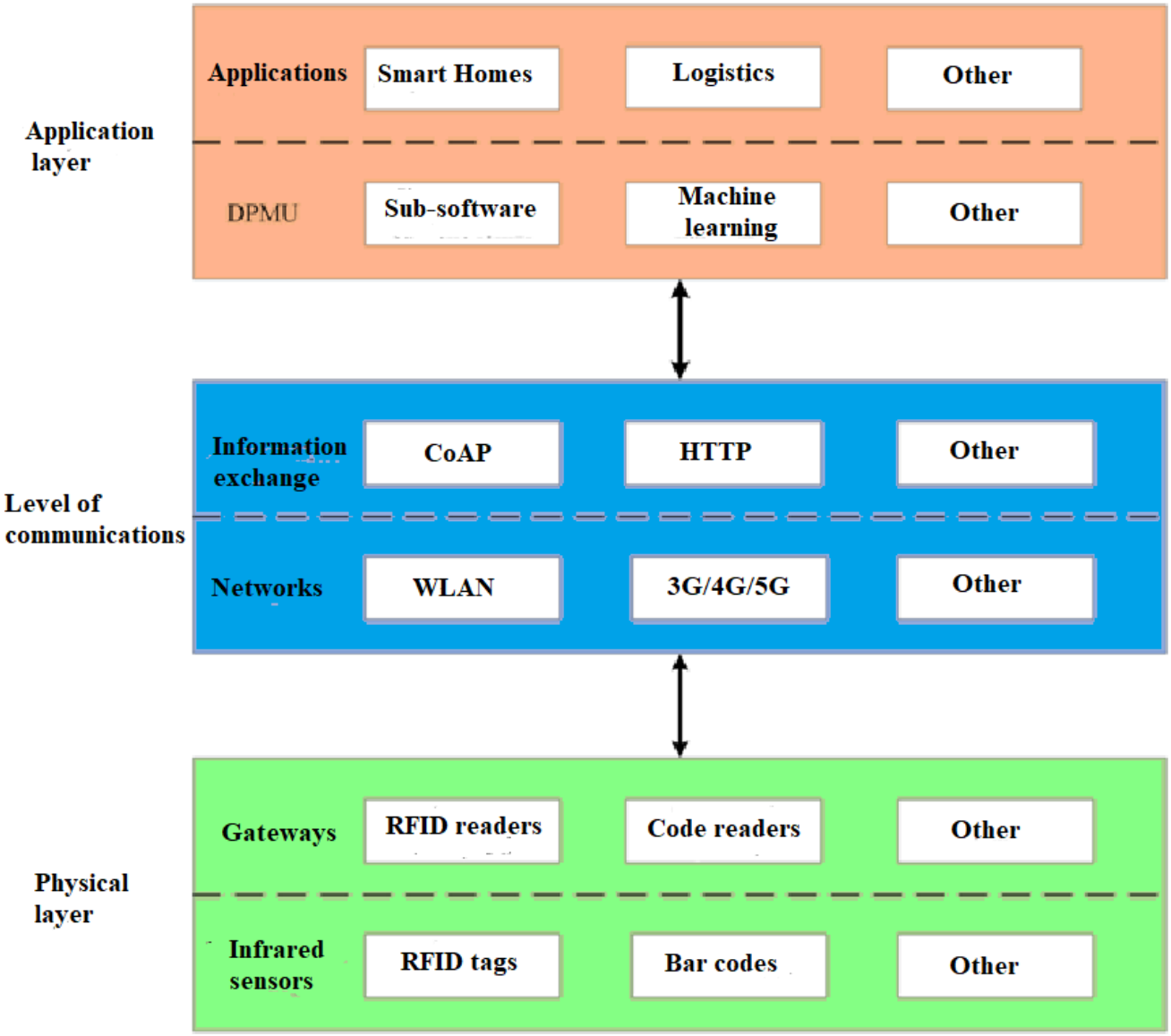

Figure 1.1 – Architecture of the Internet of Things

Typically, the data collected by IoT is sent through wireless sensors and then sent to the end user. Such sensors are very small and have limited computing power. This is done to ensure low power consumption. Thus, the communication layer provides communication between various Internet of Things and provides routing through the network gateway to the end device in the system, which collects and processes the received information.

The highest layer of the layer stack in IoT is the application layer. At this level, the data obtained from the previous two levels are processed, analyzed, stored in a database and formed into data that can be perceived by a person using various types of applications, depending on the end user's goals.

1.2 Areas of application of the Internet of Things and risks associated with security issues of data transmission protocols

If we consider the field of application of iot devices in more detail, it becomes clear that this is a huge number of devices that are used in an infinite number of areas of human activity. IoT can be found in any industry that needs to be automated. For example, in those industries where it is necessary to monitor the state of objects or devices. Implementation of automated analysis of the state of devices or objects with the help of IoT will make it possible to monitor the performance of the system in real time and adjust the relevant nodes only when necessary, which saves the time of qualified specialists and the resources of the relevant city or private organizations.

We see that the use of IoT both in everyday life and on a national scale can bring enormous benefits and improve people's lives. However, it is worth paying attention to what data is transferred between devices, what protocols are used for this and how protected it is. After all, along with the huge benefits that IoT technologies can bring, there are certain risks that the data to which the corresponding devices have access can be intercepted or obtained by attackers. In worse cases, attackers can try to take control of these devices and start giving them commands that suit the attackers' interests. That is why neglecting safety rules in matters of using IoT can lead to bad consequences. All these points must be taken into account when using this technology in practice. It is necessary to build this system in such a way that it is protected and safe.

1.2.1 Use of Internet of Things technology in urban projects

Projects that can be implemented in large cities with the help of IoT give endless space for imagination. After all, there are many things and processes in the city that need to be automated or analyzed. In particular, the use of similar technologies is possible in matters related to urban transport. You can track exactly where the trolleybus is and notify the user of its arrival time.

You can monitor the flow of people moving through the city and plan the routes and frequency of public transport in such a way that it is as efficient as possible.

1.2.2 Use of Internet of Things technology in the agricultural sector

In particular, the use of sensors that monitor the condition of the soil in the fields will make the processing of these fields, their fertilization and irrigation more efficient.

The use of drones and the analysis of the data collected by them will allow to assess the current state of the crop, and to take appropriate measures if necessary. This will greatly simplify the control process. After all, the farmer will receive data almost instantly, and will not need to walk around the field and check everything himself.

1.2.3 Use of Internet of Things technology in the construction industry

A very promising and popular direction is the use of IoT in the construction industry and in the design of houses. In particular, the presence of "smart" light or water meters, sensors on communications inside buildings, heating and lighting control systems. Even connecting the elevator to this system can help to notify about breakdowns, or give an advance warning in the event that negative processes occur with the technical condition of the equipment.

1.2.4 Use of Internet of Things technology in logistics systems

The field of logistics is already quite actively using IoT in its work, because the ability to track the movement of trucks, ships or trains with postal shipments or other cargo has already become a certain industry standard.

1.2.5 Medicine and Internet of Things technologies

The use of medical devices that have an Internet connection is a very important area that allows saving a large number of lives. After all, round-the-clock monitoring of the patient's condition, which will show and analyze the amount of oxygen in the blood, heart rate, etc., will allow to react immediately and save life in case of problems. Such devices are becoming increasingly common in the private lives of people who are trying to protect their loved ones and relatives from sudden illnesses.

1.2.6 Use of Internet of Things technology in the field of security systems

Modern security systems in organizations and cities consist of many different elements, in particular, a large number of cameras on the streets of cities, in shops, in the corridors of buildings, which monitor and analyze the current situation and can capture an intruder. Also, various sensors are often connected to such systems, which react to movement, emissions of harmful substances, burning, flooding, etc.

1.3 Cloud technologies and IoT technologies

Cloud computing in IoT increases the efficiency of solving everyday tasks. Both technologies complement each other. On the one hand, IoT generates large volumes of data, and on the other hand, cloud computing accelerates the process of processing this data.

Many cloud providers use this integration to provide a pay-as-you-go model where customers pay only for specific resources used. In addition, cloud hosting as a service adds value to IoT startups by providing savings by reducing the overall cost structure.

In addition to this, cloud computing also provides better collaboration for developers, which is common in the IoT space. Because developers can store and access data remotely, the cloud allows developers to implement projects without delays. In addition, by storing data in the cloud, iot companies can access a huge amount of data.

Figure 1.2 shows the architecture of the interaction of IoT technologies with the cloud through the global network.

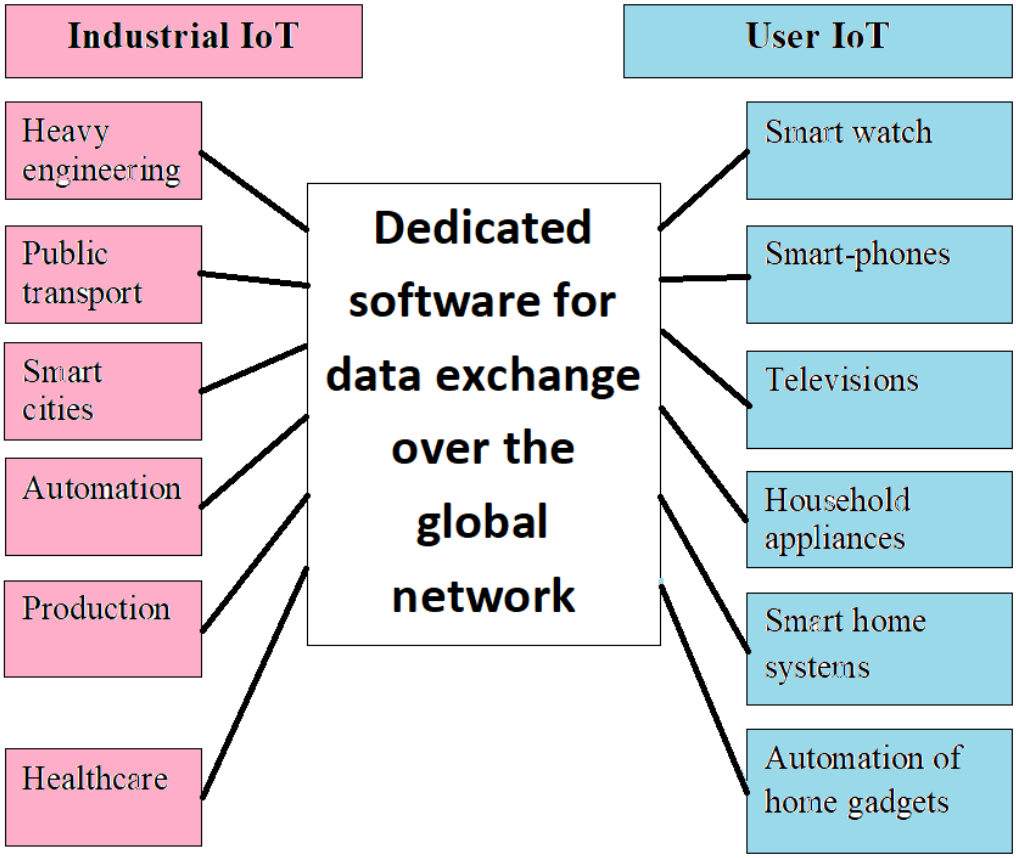

Figure 1.2 - Architecture of the interaction of IoT technologies with the cloud through the global network

The Industrial Internet of Things - IIoT (Fig. 1.2) is a set of technologies and strategies based on IoT and applied in industrial production systems. For example, iiot can be used to monitor a production system to ensure a higher level of quality. The use of IIoT is expected to have a major impact on production systems. The idea behind iiot is to use intelligent machines in conjunction with sensors to improve product quality and efficiency. An important aspect is represented by machine-to-machine (M2M) technology.

An IoT device is a smart object with internet connectivity. This is a small link in the ecosystem of the Internet faith. Thousands of interconnected devices create this ecosystem. The device could be, for example, an Internet-connected sensor or a wearable that we currently use to monitor our health. Alternatively, the object could be a simple household device that connects to the internet, or an iot sensor that we use to monitor physical properties.

The IoT platform is an important component in the IoT ecosystem. The IoT platform provides several services and is the link between devices and data stored in the cloud. The IoT platform provides several services such as:

- data store. The information sent by the sensors is stored in the cloud and used later;
- connection services: multiple devices can be connected to the iot platform using multiple protocols (MQTT, HTTP, COAP, etc.);
- data analytics: it is a set of services (using stored data) ranging from simple services to complex services such as machine learning, etc.;
- data visualization: provides several ways to present data using charts;
- gateway IoT. A gateway is an entity that connects multiple devices to the cloud. These devices can be placed in different areas, and instead of connecting directly to the cloud, the devices use a gateway to connect to it. The gateway can provide protocol translation function, for example, the gateway connects to devices using a low power protocol and sends and receives data from the cloud using a different protocol. In addition, the gateway can provide security by providing an encryption mechanism.

The IoT board is an MCU (microcontroller) that provides connections for sensors, actuators, and so on. Usually they can be connected to the Internet initially, or they can be added. For example, Arduino boards are widely used in IoT. There are boards with different functions and characteristics. We need to choose the right one according to our needs and use cases.

IoT devices generate huge amounts of data, such as engine temperature or whether a door is open or closed or smart meter readings. All this IoT data must be collected, stored and analyzed. One of the ways companies can get the most value is by pairing with an artificial intelligence system that will use IoT data and process it to make predictions.

The scheme of interaction between cloud technologies and IoT is shown in Figure 1.3 [2].

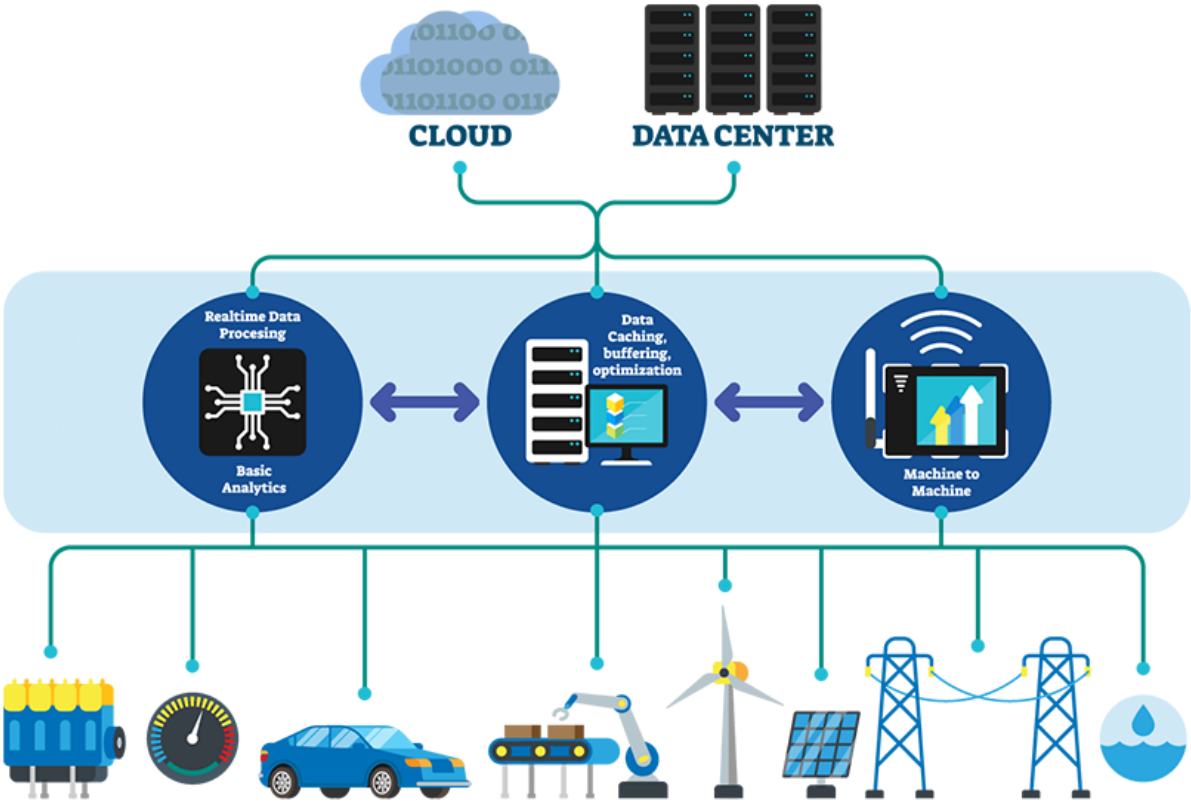

Figure 1.3 - Scheme of interaction between cloud technologies and IoT

Cloud computing is based on the principles of speed and scale. In turn, IoT applications are built on the principle of mobility and widespread networks. Therefore, it is very important that both the cloud and IoT form IoT cloud applications, aiming to make the most of their combination. Here are some reasons why the cloud is important in terms of IoT success.

1. The cloud provides remote computing power. Cloud as a technology allows iot to go beyond conventional devices such as air conditioners, refrigerators, etc., this is because the cloud has such extensive storage that it eliminates dependencies on local in-

frastructure. With the rise of miniaturization and the transition of 4G to faster internet speeds, the cloud will allow developers to offload fast computing processes.

2. Cloud provides security and privacy. The role of IoT in using mobility is huge. However, its capabilities would be incomplete without security. The cloud has made IoT more secure through preventive, detective, and corrective controls. It provided users with strong security measures by providing efficient authentication and encryption protocols. In addition to this, it has become possible for IoT products to manage and protect the identity of users using biometrics. All this is possible thanks to cloud security.

3. The cloud removes the entry barrier for hosting providers. Today, many IoT innovations are focused on plug-and-play hosting services. That's why the cloud is perfect for IoT. Hosting providers should not depend on massive hardware, or even any other hardware that will not support the flexibility of IoT devices. The cloud allows most hosting providers to provide their customers with a ready-made model, removing entry barriers for them.

4. The cloud facilitates communication between devices. The cloud acts as an intermediary in communication when it comes to IoT. Many powerful APIs such as Cloudflare, cloudcache, and Dropstr are supported by cloud connectivity, making it easy to connect to smartphones. This allows devices to communicate with each other, and not just with us, which is essentially the principle of the Internet of Things cloud.

It would be fair to say that the cloud can accelerate the growth of IoT. However, the deployment of cloud technologies also has certain problems and disadvantages. Not because the cloud is a technological bug, but the combination of the IoT cloud can burden users with some hurdles. If you ever use an IoT cloud solution, it's best to know ahead of time what issues you might run into.

1.4 Difficulties in integrating cloud technologies and the Internet of things

Along with the positive results of integrating cloud technologies and IoT, a number of difficulties arise.

1. Processing a large amount of data. Handling large amounts of data can be very difficult, esp. When there are millions of devices in the system. This is because there is an overall application performance issue. Hence nosql's motion tracking can be useful, but it's untested and untested in the long run. This is why there is no reliable method for managing large amounts of data for the cloud.

2. Network and communication protocols. The cloud and the Internet of Things involve machine-to-machine communication between many different types of devices having different protocols. Managing this kind of change can be difficult since most of the application areas are not related to mobility. Wi-Fi and Bluetooth are currently being used as an access restriction solution to facilitate mobility to a certain extent.

3. Sensor networks. Sensor networks have enhanced the benefits of IoT. These networks allow users to measure, analyze and understand sensitive indicators from the environment. However, processing the large amount of data from this sensor in a timely manner was a major challenge. The cloud provides a new opportunity for aggregating sensor data, but also hinders progress due to security and privacy concerns.

1.5 Cloud computing technology

Cloud computing is a technology that allows the user to use remote resources and capacities. The calculation is based on the principle of distributed data processing. Its essence is to provide the user with remote access to services provided in the cloud. The need to save money due to effective measures for the provision of services in the field of hosting influenced the development of this technology.

Today, the term "cloud computing" refers to a set of various services that can be accessed via the Internet. In fig. 1.4 we can see the architecture of cloud services.

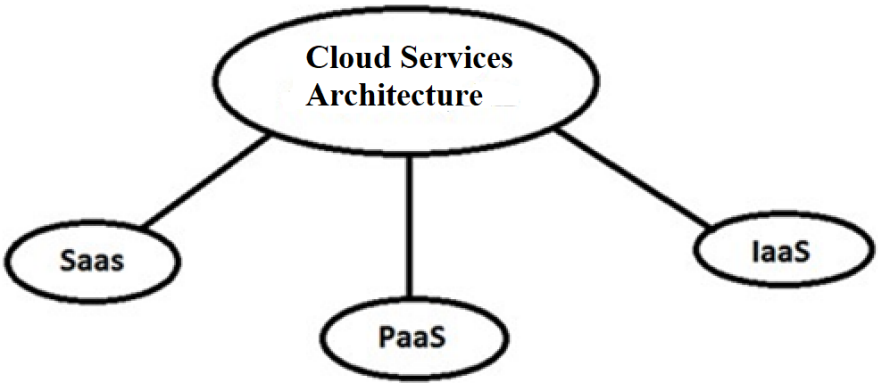

Figure 1.4 – Architecture of cloud services

SaaS is a cloud application service, probably the most popular and easy to use form of cloud computing. Saas uses the Internet to deliver applications that are managed by third-party vendors and whose

The interface is available to the client side. Most Saas applications can be run directly from a web browser, without the need for downloads or pre-installation. SaaS eliminates the need to install and run applications on personal computers. With the use of SaaS, the task of enterprises for rational maintenance and support is simplified. The provider's services include maintenance of: applications, runtime, data, middleware, operating systems, server virtualization, storage and networks. Gmail is one well-known example of a Saas mail provider [2].

PaaS is the most complex of the three types - a cloud service platform that distributes computing resources through a separate platform. Developers get an option with Paas where they can build their apps or customize apps. Paas makes development, testing and deployment of applications faster, easier and cost-effective, saving the user from the need to purchase lower layers of hardware and software. One difference between Saas and Paas is that some aspects of Paas are managed by users rather than providers.

PaaS provides computing infrastructures, hardware and platforms that are installed on top of the hardware. In the same way that you can create macros in Excel, PaaS allows you to create applications using software components managed by a third-party vendor. Paas scales well and users don't have to worry about platform updates or their site going down during maintenance. The users who get the most benefit from PaaS are companies that want to increase the efficiency and interactivity of a large staff of employees [3].

Iaas is a cloud infrastructure service that provides computing infrastructure (for example, an environment virtualization platform), storage, and networking. Instead of buying software, servers or network equipment, the user can buy all this as a completely external service, the cost of which usually depends on the amount of resources consumed. In other words, a third party, for a rental fee, allows the user to install a virtual server on their IT infrastructure. Compared to SaaS and PaaS, Iaas users have a lot of responsibility for managing: applications, data, runtime, middleware and operating systems. Service providers still control virtualization, servers, hard drives, storage, and networking. IaaS users get full access to the ready-made information infrastructure, in the middle of which they can install the necessary platforms. Users are responsible for updating if new versions of the platforms fail.

1.6 Pros and cons of using cloud technologies

The advantages of cloud technologies include.

1) Relatively cheap computers for users. There is no need to buy high-power computers with a large amount of memory and disks, because all information and all programs are stored on servers in "clouds" and run remotely. From large stationary personal computers and ordinary laptops, users can switch to compact netbooks.

2) The productivity of computers for users increases. Due to the remote launch of files and programs, users' computers are not burdened with this work and function faster. As an example, you can consider the work of Panda Cloud Antivirus - an antivirus program available as a Web server. Panda Cloud Antivirus provides the ability to scan data for viruses remotely on powerful servers. Running the same program directly on the user's computer using his own resources would increase the load by approximately 2 times.

3) The efficiency of using the IT infrastructure increases and costs decrease. If we take the average estimate of server loading for the company, it will be about 13%. Sometimes it becomes necessary for the company to use additional capacities, but most of the time computing resources are not busy. If you use computing resources on remote servers in the "cloud", the company's expenses for this part can be halved. Taking into account the changing economic situation, the flexibility of production is increasing. Companies that do not trust the storage of their data to third-party organizations have the opportunity to build their own cloud and receive all the benefits of infrastructure virtualization.

4) Reduction of costs for maintenance and purchase of software. With the use of Cloud Computing technology, companies have fewer of their own servers, so it becomes easier to service them. With the decrease in the number of physical servers, problems with the purchase of software framework provision are also decreasing. Due to the fact that services and applications are in the "cloud", there is no need to buy software for each user. The company buys the necessary programs in the "cloud". The cost of services offered with access via the Internet is an order of magnitude lower than analogs that exist for personal computers. Moreover, it is possible to rent the use of programs by the hour, and the costs of maintaining the software in working condition and updating it are zero.

5) Growth of computing power. Compared to a personal computer, the computing resources of the cloud provide enormous opportunities. The computing power of the cloud is determined by the number of its servers. In other words, users are provided with remote access to a supercomputer, which provides an opportunity to solve more complex and voluminous tasks that cannot be handled by a regular PC.

6) Unlimited volume for data storage. The amount of space provided in the cloud for data storage can be flexibly and automatically adjusted, depending on the user's wishes. If the usual situation for a personal computer user is a lack of space for data, it is practically impossible for a cloud computing user.

7) Compatibility with operating systems. For cloud technologies, it does not matter what operating system the user has. A Microsoft Windows client can easily exchange data through the cloud with Unix users. As for access to cloud computing services, they are provided using standard browsers for each operating system.

8) Compatibility of document formats. In cloud computing, there is no concept of incompatibility of document formats created in different versions of the same cloud program.

9) Ease of collaboration for a group of users. A convenient possibility of simultaneous work of several participants appears in the cloud computing system. There is no need to transfer documents from computer to computer. Editing of documents is reflected instantly, and the user always has access to the latest version of the updated document.

10) Ubiquitous access to user files is a big advantage of using cloud computing. If the data is stored in the cloud, it is always available to its owner if there is access to the Internet. The user also has a wide choice of different devices from which he can access this. The cloud client can use a personal computer, netbook, laptop, tablet computer, smartphone.

11) Reducing the use of natural resources. With cloud computing technology, savings are not only on electricity, computing power and physical space, but also on natural resources. Data processing centers can be located deliberately in cold climates. Data access equipment is now more compact, requiring fewer materials to manufacture.

12) Data resistance to loss. Data stored in the cloud distributes its copies over several servers. The probability of data loss in the cloud is much lower than the probability of data loss when stored on ordinary physical data carriers.

Disadvantages of cloud technologies include.

1) The constant need to connect to the Internet. Cloud computing always requires a connection to the Internet. There are, of course, a number of applications that are downloaded to the computer and allow further work regardless of the connection. In other cases, everything is simple: no connection - no work. According to many, this is the biggest drawback of cloud computing. But if we take into account the development of information technologies in our time, then we can safely say that access to the Internet is almost everywhere. Therefore, soon this problem will completely disappear.

2) Works poorly with a slow connection. Most cloud services require a fast Internet connection for normal operation. But as mentioned above, information technologies do not stand still, so there are currently no problems with the bandwidth of Internet connections.

3) Programs may run slowly and with incomplete functionality. Some of the programs provided by cloud services run faster on a local computer. This may be due to both the low bandwidth of the Internet connection and the load on remote servers. Also, programs presented in the cloud have limited functionality, unlike their versions for a local computer.

4) There is a threat to data security. Of course, if you transfer data to the cloud, then there is an immediate possibility of a threat to the security of information. But here the whole point is trust in the provider. If the cloud technology provider reliably encrypts information transmission, creates backup copies and already has a lot of experience in the market in this area, then a security breach may never happen.

5) The fact that data loss in the cloud is irreversible. But it is much more difficult to bring them to such a state than to lose them on a local computer.

6) Despite the many advantages over disadvantages, everything happens differently in each situation. Co the woman herself evaluates all the criteria and makes a choice: to use cloud computing or not [2].

2 FOG COMPUTING AND THE INTERNET OF THINGS

Fog Computing - a term coined by Cisco in 2012, also known as "fogging", essentially means extending computing to the edge of an enterprise network, instead of hosting and operating from a centralized cloud. This facilitates local data processing in intelligent devices and smooths the operation of computing, storage and network services between end devices and cloud computing data centers.

Fog computing supports IoT, 5G, artificial intelligence (AI), and other applications that require ultra-low latency, high network bandwidth, resource constraints, and added security.

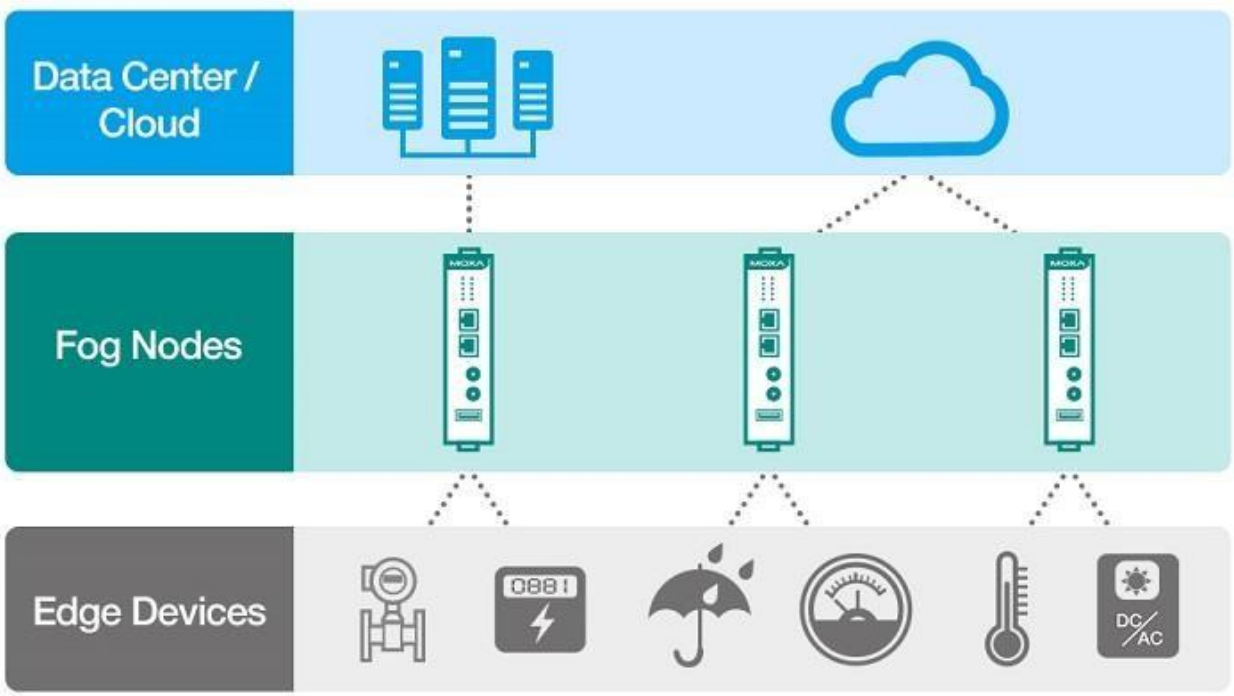

Figure 2.1 - Structural diagram of Fog Computing

Both Cloud and Fog Computing use similar IT resources: computing devices (servers and processors of users' computers), network switching nodes and data storage systems. However, extending the cloud to network boundaries is not just about scaling that cloud.

2.1 The difference between fuzzy and boundary computing

Peripheral (edge) computing is an approach related to data processing at the edge of the network, where data is created, rather than in a centralized storage dedicated to data processing.

Edge computing systems are a distributed open IT architecture that uses decentralized processing and supports mobile computing technologies of the Internet of Things. When using peripheral computing, the data is processed by the device itself, a local computer or server, and not transmitted to a data center.

Edge computing systems provide acceleration of data flows, including real-time data processing without delay. They allow intelligent applications and devices to respond to data almost immediately after it is created, eliminating any delays. This is critical to the development of technologies such as self-driving cars, and provides significant benefits for organizations.

Edge computing provides efficient processing of large volumes of data close to the source, reducing the load on Internet channels. On the one hand, it allows you to reduce costs, and on the other hand, it allows you to effectively use programs remotely. In addition, the ability to process data without placing it in the public cloud provides an additional level of protection for confidential data.

In turn, fuzzy computing always uses boundary computing, but not vice versa. Fogging is a system-level architecture that provides tools for distributing, orchestrating, managing, and securing resources and services across networks and between devices at the edge.

Edge computing architectures place servers, applications, or small clouds at the edge. Fog computing has a hierarchical and flat architecture with multiple layers that form a network, while edge computing relies on individual nodes that do not form a network.

Fog computing has extensive peer-to-peer communication between nodes, where each edge runs its nodes in silos (fragments of a network isolated from each other), requiring data to be transported through the cloud for peer-to-peer traffic. It is also worth noting that fog computing includes the use of cloud services, while edge computing excludes their use altogether.

2.2 The main architectural differences between Fog and Cloud

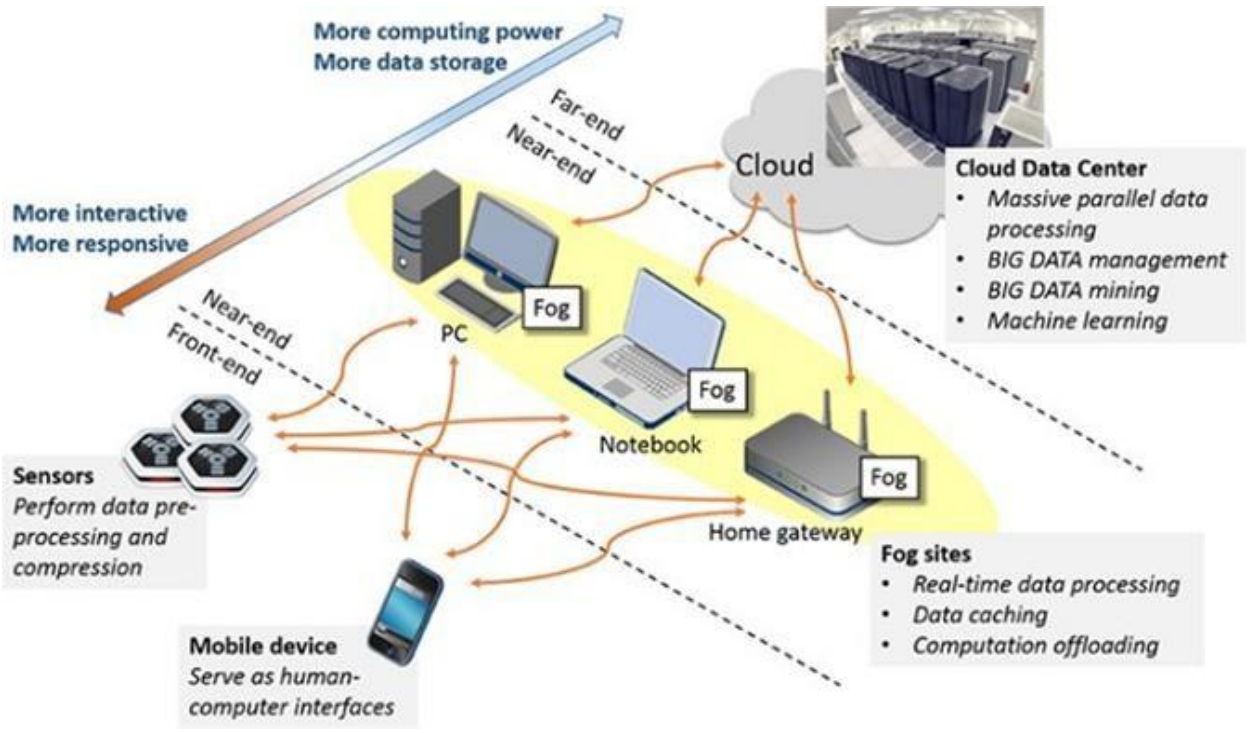

Figure 2.2 - Fog Computing network architecture [2]

The main difference between fog computing and cloud computing is that cloud is a centralized system, while fog is essentially a distributed decentralized infrastructure.

Fog knots:

- Receive feedback from IoT devices using any protocol in real time.

- Run IoT applications for real-time monitoring and analysis, with millisecond response times.

- Provide temporary data storage, usually 1-2 hours.

- Send periodic data summaries to the cloud. Cloud platform:

- Receives and aggregates data summaries from multiple fog nodes.

- Performs analysis of IoT data and data from other sources to obtain business information.

- Can send new application rules to fog nodes based on this data.

The main architectural differences between Fog and Cloud are:

- ensuring the quality of services (qos, Quality of Service), which requires dynamic adaptation of applications to the state of the network;

- tracking the location (Location Awareness) in order to maintain the stability of the application in conditions of terminal mobility;

- tracking of contextual information (Context Awareness), that is, the ability to detect the presence of available resources nearby in order to use them in the work of the application, with the possibility of horizontal interaction.

In the Fog architecture, network nodes (Fog Sites) are located closer to cloud data centers and have greater computing power with a larger volume of data in storage systems. Network nodes located closer to IoT sensors and mobile devices have greater interactivity and faster response. A distinctive feature of Fog is that user devices such as personal computers, home gateways, set-top boxes and mobile devices can act as a network node. In order for the user's device to work as a node of the Fog network, the user must give the operator the appropriate permission to use the computing power of his gadget in the background, in exchange for various benefits from the operator.

Developers port or write IoT applications for nodes at the edge of the network. Fog nodes closest to the edge of the network receive data from IoT devices. Then - and this is extremely important - fog IoT applications direct different types of data to the optimal place for their analysis.

- In most cases, time-sensitive data is analyzed on the fog node closest to the data generation sources.

- A centralized aggregation cluster (node) is used to analyze data that can wait for seconds or minutes, after which an action is performed.

- The least time-sensitive data is sent to the cloud for data analysis and storage. An example here can be each of the fog nodes sending periodic summaries of data to the cloud for their analysis.

2.3 Examples of Fog Computing applications

Autonomous driving systems (ADS, Autonomous Driving System). ADS uses various multi-mode sensors, computer vision and image analysis technologies, satellite and network positioning on maps and intelligent analytics, on the basis of which ADS helps to control the driver or controls the self-driving vehicle. In such applications, high

speed is required, so the Fog-node with artificial intelligence elements must be placed directly in the vehicle.

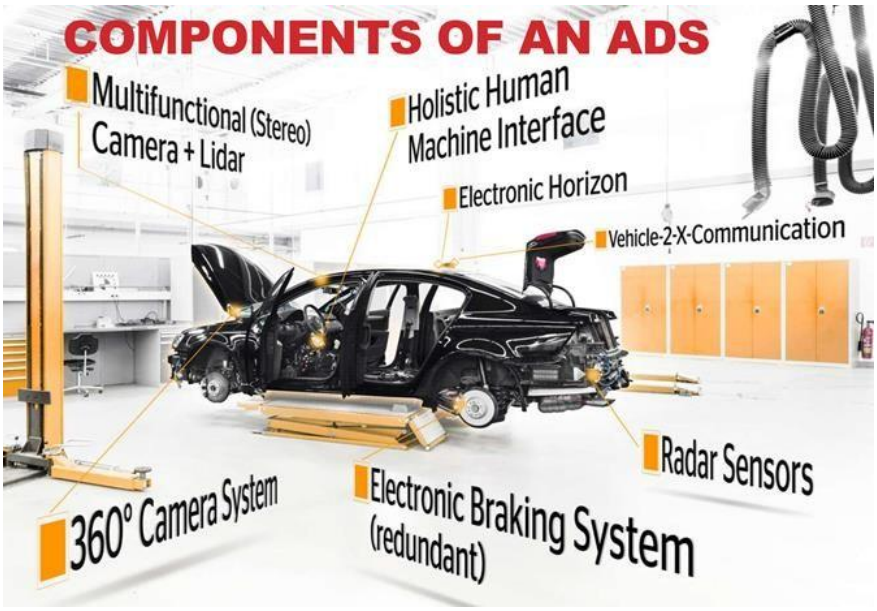

Figure 2.3 - Components of Fog nodes in ADS [2]

Fog-systems in electronic healthcare (e-health). Fog-systems in medicine are used in those cases when it is necessary to carry out an operational analysis of the data received from the sensors worn by the patient and to take immediate actions in accordance with the treatment plan.

For example, in some places, Fog-technologies are already used to control the condition of diabetic patients and to automatically administer injections. A sensor on the patient's body determines the critical value of the sugar content in the blood, and through the Fog network issues a signal to perform an injection using a microsyringe also located on the patient's body. Thus, the patient gets rid of the need to constantly make measurements and injections himself.

Fog-projects of cloud providers. In 2016, the three largest providers of cloud platforms - Amazon, Google and Microsoft - started several projects using Fog Computing in their iot ecosystems, in which the so-called "Serverless architecture" (serverless architecture).

The serverless architecture allows you to execute the source code of thousands and millions of users (in particular, fog devices) inside the computing environment, without worrying about scaling resources.

Microsoft has announced support for Azure Functions (Azure Functions) within the SDK (Software Development Kit) development platform.

Azure functions were initially introduced in the family of cloud products with serverless architecture (Serverless Architecture), developed at Microsoft.

Amazon has developed the Greengrass platform with support for the so-called Lambda functions (serverless architecture) in iot devices when interacting with the AWS cloud platform. Greengrass is a software module execution container that can be run directly on the Fog device, rather than on a server in a data center. Devices with Greengrass can exchange information with each other regardless of the presence of the external Internet, that is, horizontally between Fog devices using various radio protocols of the Internet of Things.

Google introduced the Android Things Internet of Things platform with support for Intel Edison and Joule 570x microcomputers, NXP Pico i.MX6UL and Argon i.MX6UL, as well as Raspberry Pi 3. Fog applications are developed on the Android Studio platform for any of these devices Android Things also provides integration with Google Play and the entire Android ecosystem, which currently powers 90% of the world's smartphones. Thus, the Android Things system enables any Android smartphone or tablet to work as a Fog node.

3 ANALYSIS OF VULNERABILITY OF IoT NETWORKS USING CLOUD TECHNOLOGIES

3.1 Classification of threats of cloud services

Examples of the main attacks on cloud services are given below.

1) Traditional attacks on software. They are related to the vulnerability of network protocols, operating systems, modular components and others. To protect them, it is enough to install an antivirus, firewall, IPS and other components. It is only important that these protection tools are adapted to the cloud infrastructure and work effectively in virtualization conditions.

2) Functional attacks on cloud elements. The type of attacks is related to cloud layering, a general security principle where the overall protection of the system is equal to the protection of the weakest link. Thus, a successful Denial of Service (DOS) attack on a proxy server installed in front of the cloud will block access to the entire cloud, despite the fact that inside the cloud all connections will work without interference. Similarly, an SQL injection passed through the application server will give access to system data, regardless of the access rules in the data storage layer [7]. To protect against functional attacks for each layer of the cloud, it is necessary to use specific means of protection: for the proxy - protection against dos attacks, for the web server - page integrity control, for the application server - application level screen, for the DBMS layer - protection against SQL injections, for the storage system – backup and demarcation of access. Individually, each of these defense mechanisms is already in place, but they are not assembled together to provide comprehensive cloud protection.

3) Attacks on the client. The type of attack has been developed in the web environment, but it is also relevant for the cloud, since clients connect to the cloud, as a rule, using a browser. Attacks such as Cross Site Scripting (XSS), interception of web sessions, password theft, "man in the middle" and others fall into it. Defense against these attacks traditionally consists of strict authentication and the use of an encrypted connection with mutual authentication, but this is an expensive solution. Therefore, there are still unsolved tasks in this field of information security.

4) Threats of virtualization. These threats occur because the platform for cloud components has traditionally been virtual environments. Such attacks on the system also threaten the cloud as a whole. Solutions are now beginning to appear for some virtualization threats, but this industry is quite new, so no permanent solutions have been developed yet.

5) Complex threats to clouds. Cloud control and management is also a security concern. How to ensure that all cloud resources are accounted for and that there are no uncontrolled virtual machines in it, unnecessary business processes are not launched, and the mutual configuration of cloud layers and elements is not violated. This type of threat is related to the management of the cloud as a single information system and the search for abuses or other violations in the operation of the cloud, which can lead to unnecessary costs for maintaining the functionality of the information system. For example, if there is a cloud in which a virus may exist, how to prevent information theft. This type of threat is the most multi-level, for which it is impossible to find a universal means of protection. Protection must be built individually for each cloud.

The first two types of threats have already been sufficiently studied and defenses have been developed for them, but they still need to be adapted for use in the cloud. For example, the firewall is designed to protect the perimeter, but in the cloud it is not easy to allocate a perimeter for an individual client, which makes protection much more difficult. Therefore, the technology needs to be adapted to the cloud infrastructure.

A new type of threat for cloud services is virtualization problems. The fact is that when using this technology, additional elements appear in the system that can be attacked. These include a hypervisor, a system for transferring virtual machines from one node to another, and a virtual machine management system.

Let's consider in more detail what kind of attacks the listed elements can be subjected to [6,7].

1) Attacks on the hypervisor. The key element of the virtual system is the hypervisor, which ensures the division of physical computer resources between virtual machines. Interference with the hypervisor can lead to the fact that one virtual machine can gain access to the memory and resources of another, intercept its network traffic, take away its physical resources, and even completely oust the virtual machine from the server. So far, few hackers understand exactly how the hypervisor works, so there are practically no attacks of this type, but this does not guarantee that they will not appear in the future.

2) Migration of virtual machines. A virtual machine is a file that can be run on different cloud nodes. Virtual machine management systems provide mechanisms for transferring virtual machines from one node to another. However, it is possible to steal the virtual machine file and try to run it outside the cloud. It is impossible to take out a physical server, but a virtual machine can be stolen over the network without having physical access to the servers. True, a separate virtual machine outside the cloud has no practical value, dearyou need at least one virtual machine from each layer, as well as data from the storage system to restore a similar cloud, however, virtualization allows the theft of parts or the entire cloud as a whole. That is, interference with the mechanisms of transfer of virtual machines creates new risks for the information system.

3) Attacks on control systems. The huge number of virtual machines used in clouds, especially in public clouds, requires such management systems that can reliably control the creation, migration and disposal of virtual machines. Interference with management systems can lead to the appearance of invisible virtual machines, the blocking of some machines and the substitution of unauthorized elements in the layers of the cloud. All this allows attackers to obtain information from the cloud or capture parts of it or the entire cloud.

It should be noted that so far all the threats listed above are purely hypothetical, since there is practically no information about real attacks of this type. At the same time, when virtualization and clouds become quite popular, all these types of attacks may turn out to be quite real. Therefore, they should be kept in mind even at the stage of designing cloud systems.

3.2 The main vulnerabilities of Internet of Things networks

3.2.1 Danger of physical access

Before designing such systems, it is necessary to consider how difficult it will be for an attacker to gain direct access to a given sensor or device. How this device stores data, whether it is encrypted, and whether it can be accessed. What interfaces the device has, and whether they would allow an attacker to connect and take control, etc.

3.2.2 Danger of default settings

Considering the prevalence of IoT devices in people's lives, and considering the situation where users do not pay enough attention to the security settings of their devic-

es, it becomes clear that there is a big problem related to unsafe settings that are installed by the manufacturer at the factory. This can be the absence of a password for connection, the absence of data encryption, the use of insecure network communication protocols, etc. That is why, after connecting the device for the first time, it is necessary to check the default settings and set them in such a way that the device is protected.

3.2.3 Limited Support and Updates

Since a lot of IoT devices are produced, they have separate ecosystems with different standards, not every manufacturer can perform scheduled firmware updates and organize decommissioning of outdated devices. That is why, sometimes, one and the same model, using different firmware, may or may not have vulnerabilities.

In order to protect yourself, you need to follow the terms of support for this or that device, follow the release of security updates, and install new firmware versions in a timely manner, which have all security updates and make the use of this device safer.

3.2.4 Unsecured transmission and storage of data

There is a risk associated with the storage and transmission of data.

Most manufacturers try to store data in devices securely, but often forget about security when transferring data, and may use unsecured wireless connections and use unsecured data transfer protocols. It is also worth paying attention to when choosing a more secure protocol.

3.2.5 Inadequate Privacy Protection

This point follows from the previous one and is directly related to the fact that often devices, in particular fitness trackers or "smart" watches, collect a huge amount of confidential user information, and at the same time may have security problems when storing it and transferring it to other devices for further processing

3.2.6 Use of no Legacy Components

Any device manufacturer is trying to save money, produce more devices at a lower price, and thus get more users into their ecosystem. On the other hand, the manufacturer may not be able to create an entire firmware for their iot devices from scratch, so they will use outdated software, free libraries that may have vulnerabilities, etc.

3.2.7 Lack of secure update mechanisms

As discussed above, certain IoT devices may have some software security issues due to the use of outdated libraries. In addition, most users do not follow the procedure of updating their devices. Not all IoT devices have secure software and firmware update mechanisms.

3.2.8 Dangerous interfaces in the device family ecosystem

A very common problem is that manufacturers of IoT devices can save on the security of API interfaces when building an ecosystem of device families. At the same time, not enough attention is paid to the issue of authentication. This is very dangerous, because it can lead to the fact that the device in the ecosystem can be compromised. And the lack of encryption when accessing API interfaces can create a large number of problems related to unauthorized access to data.

3.2.9 Unsafe network services

Often, when designing firmware and software, insufficient attention may be paid to the use of system services. Especially when using a large number of open unprotected libraries or components. In particular, this may lead to the case when there may be active services in the system that are constantly connected to the Internet and perform data processing and transmission.

3.2.10 Bad Default Passwords

Manufacturers often set default passwords for the user to change at first launch, but often users may not do so.

Another problem is that users can use the same passwords on many sites and systems.

3.3 Overview of the main protocols of Internet of Things technology and their vulnerabilities

3.3.1 DDS protocol

Let's consider the Data Distribution Service (DDS) protocol in more detail. To do this, consider a case in which it will be appropriate to apply it. Namely, consider the connection between two sensor nodes. To do this, we designate the segment of the net-

work between the sensor nodes as section number 1. A number of tasks are performed on this section, for example, the distribution of information between sensor nodes for temporary storage or forwarding. In order to ensure communication between sensor nodes or sensors, the DDS protocol mentioned above is used.

The DDS protocol implements two operations: reading and writing. Let's pay attention to the fact that the write operation is quite primitive, so let's focus on the read operation. The read operation is performed on all available devices. Data is not removed from the local DDS cache as a result of this operation and can be read again using special parameters.

Therefore, the DDS protocol distributes data between devices. DDS implements direct bus communication between devices taking into account the relational data model. The DDS protocol implements a multicast system using UDP. This protocol is based on the publisher-subscriber pattern, while message transmission is carried out over the bus using the request-response method.

After research by cyber security specialists of the DDS standard in its various manifestations, more than a dozen different vulnerabilities were found. Among the vulnerabilities that were found were:

- network amplification;
- incorrect method of processing an invalid structure;
- possibility of buffer overflow.

The researchers emphasize that the DDS protocol has a large and fairly complex code base, which is extremely difficult to analyze completely. Therefore, there remains a high probability that with more thorough analysis, many more vulnerabilities will be found, and they may be more serious than those currently known.

3.3.2 COAP Protocol

The DDS protocol is quite advanced and widespread, but sometimes in IoT systems there may be certain limitations that make this protocol less attractive. In order to bypass these limitations, it becomes appropriate to use another protocol. Namely, the Constrained Application Protocol (COAP), which is specially designed for networks with limited resources and low energy consumption.

COAP is a specialized transmission protocol that was specially designed for use in resource-constrained networks and devices, M2M applications, etc. The protocol can be seen as a certain complement to the HTTPS protocol, but there are some differences.

Unlike the HTTPS protocol, the coap protocol is intended for use in devices with certain limitations. Also, it is worth noting that coap uses the UDP transport protocol.

Therefore, the COAP protocol is convenient for data transmission in networks with a large number of sensor nodes and brokers. To ensure normal communication between them for the purpose of registration and configuration of nodes.

However, the COAP protocol is not the only protocol used for data transfer, besides COAP, there is also the popular XMPP protocol for similar tasks. It is important to understand that the choice of the right protocol should depend on the task, on the operating conditions of the network and the goals of the devices from which the system is built. This is the only way to choose a really correct protocol that will have all the necessary functionality and characteristics to perform a specific task.

Unfortunately, the COAP protocol is very convenient for DDOS (Distributed Denial of Service) attacks.

Also, this protocol can be vulnerable to IP spoofing. By replacing the IP address of the sender with the IP address, it allows you to perform "dropping" with large data packets

3.3.3 XMPP protocol

Tasks such as:

- registration of the sensor node;
- configuration and setting of nodes;
- transfer and distribution of information.

As mentioned above, the COAP or Xtensible Messaging and Presence Protocol (XMPP) protocol can be used for similar tasks. Each of them is chosen in accordance with the task, because they have certain differences and each of them is better suited to each of their tasks.

XMPP is best suited for small personal networks. If we turn to its characteristics, XMPP is a protocol for exchanging messages and information about presence. The protocol is extensible.

If we consider the application of the XMPP protocol in the IoT environment, then its advantage is that it provides a simple way of addressing devices. A huge advantage of the XMPP protocol is that its addressing is particularly convenient in cases where data needs to be transferred between remote, and often independent, points, such as two separate subscribers.

XMPP has some security issues, which are related to insufficient encryption of data transmitted between users.

3.3.4 MQTT protocol

Most of the tasks related to IoT devices calmly perform the protocols that were listed above, but there is a certain set of specific tasks where something special is needed that can take into account the peculiarities of these tasks.

So, if we take into account networks with a large number of various devices, then there will definitely be an overloaded communication channel. In order to reduce the load on it, you can use the Message Queue Telemetry Transport (MQTT) protocol, which allows you to organize certain queues.

Protaround MQTT - a special protocol that is intended for use in the field of telemetry and remote monitoring and uses queue construction for this.

MQTT is a binary publisher-subscriber protocol that uses the TCP protocol for data transport.

Among the problems associated with the MQTT protocol is a huge problem related to incorrect settings. Namely, there is no login password. Since devices can be connected to the network, they can be indexed by search engines. Among the search engines, there is the Shodan system, which allows you to find similar devices.

Table 3.1 - Vulnerabilities of Internet of Things protocols

Protocol	Vulnerability
DDS	Amplification of the network, incorrect way of processing received invalid structure, possibility of buffer overflow
COAP	Vulnerable to DDOS attacks and IP spoofing
XMPP	DDOS attacks, poor validation of XMPP packets, insufficient encryption of sent data
MQQT	Weak passwords and poor settings for access rights, connecting to a public network

4 INVESTIGATING OF MULTI-LEVEL PROTECTION OF IoT NETWORKS WITH THE INTEGRATION OF FOG COMPUTING AND CLOUD COMPUTING TECHNOLOGY

4.1 Investigating of multi-stage protection

Various methods are used to protect information systems. Each method gives a certain level of probability of repelling the threat. In practice, various options for organizing protection systems are possible. To make the system cheaper, a single-stage protection system is used. But such an architecture is not always effective.

It is possible to increase the efficiency of the protection system due to the multi-level architecture of the system. At each border of the protective barrier, the overall probability of information protection increases. The more protective barriers - the higher the level of protection.

It is important to study the general function of the probability of repelling a threat with a multi-level architecture, depending on the number of involved protection boundaries and the level of probability of repelling a separate threat in a complex system.

When calculating the total probability of repelling a threat taking into account several means of protection (n_i), it should be taken into account that the presence of several separate means of protection are joint events. This means that a common event occurs only when all of these events occur.

The formula for two compatible events has the form

$$P(A+B) = P(A) + P(B) - P(A)P(B) \tag{4.1}$$

For three compatible events, you can use the formula

$$\begin{aligned} &P(A+B+C) = P(A) + P(B) + P(C) - P(A)P(B) - P(A)P(C) \\ &- P(B)P(C) + P(A)P(B)P(C). \end{aligned} \tag{4.2}$$

There is a general formula for n compatible events.

For two-level protection at $P_1(t) = P_2(t) = P_p(t)$, the calculation and family of graphs (Fig. 4.1) of the increase in the probability of protection ΔP from the probability

P_2 of the second protection element in the interval from 0 to 1 with a step of 0.1 for fixed values of the probability P_1 from 0 to 1 with a step of 0.1 were performed and a family of graphs was constructed.

For this case, formula (4.1) takes the form

$$P_c = 2P_p(t) - P_p^2(t). \tag{4.3}$$

Fig. 4.1 shows graphs of the dependence of the increase in the probability of threat reflection in relation to one-stage protection. The function $\Delta P(P_2)$ at a fixed probability value P_1 has a linear character. When the probability of protection P_1 of the first border increases, the increase in protection ΔP decreases. When $P_1 = P_2 = 1$ the probability of protection is maximum and is equal to 1. Vertical arrows show the increase ΔP for cases of equality of values P_1 and P_2. The maximum increase occurs at $P_1 = P_2 = 0{,}5$.

When the degree of protection increases, the probability of repelling the threat increases. But the calculation shows that the increase in the probability of protection decreases with each additional element of protection. On an example with the same probabilities equal to 0.5, an increase of 0.25 was calculated when using a two-stage instead of a single-stage protection, and when introducing three-stage protection, the increase was 0.125 compared to a two-stage.

It is possible to increase the efficiency of the protection system due to the multilevel architecture of the system. At each border of the protective barrier, the overall probability of information protection increases. The more protective barriers - the higher the level of protection.

The use of two-stage protection instead of one-stage protection leads to an increase in the probability of threat reflection in relation to one-stage protection by 0.25 for the same threat reflection probabilities equal to 0.5.

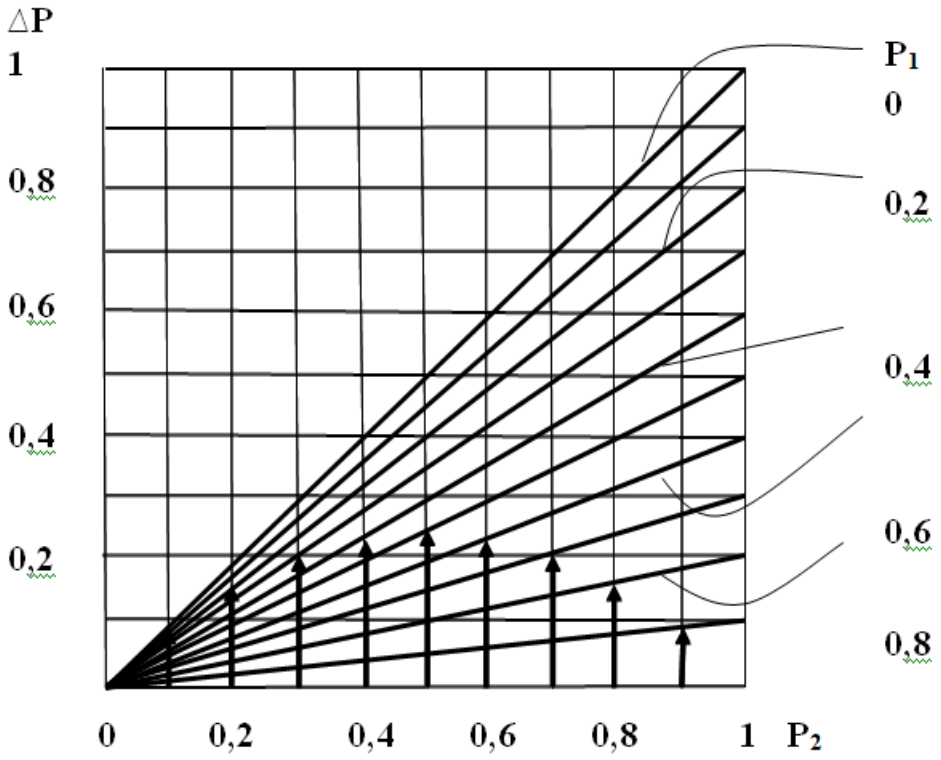

Figure 4.1 - Increased probability of threat reflection in relation to single-stage protection

The introduction of additional levels of protection leads to a further increase in the level of protection, but the level of additional protection does not increase as quickly. Thus, with the same probabilities of threat reflection equal to 0.5 at each level, the effect of increase is only 0.125 with three-level protection compared to two-level protection.

4.2 Security of cloud services regarding the use of different types of clouds

The level of risk in the three cloud models is very different and the ways to solve security issues also differ depending on the level of interaction. The security requirements remain the same, but the level of security control varies in different SaaS, PaaS, or IaaS models. From a logical point of view, nothing changes, but the possibilities of physical implementation differ radically.

In the SaaS model, the application runs on cloud infrastructure and is accessible through a web browser. The client does not have access to the network, servers, operating systems, data storage, or even some application capabilities. For this reason, in the

SaaS model, the main responsibility for ensuring security falls almost entirely on the providers.

In the SaaS model, the applications are in the cloud, so the main risk is the use of multiple accounts to access the applications. Organizations can solve this problem by unifying accounts for cloud and on-premises systems. When using the single sign-on system, users gain access to workstations and cloud services using one account. This approach reduces the likelihood of "suspended" accounts prone to unauthorized use after employees are fired.

PaaS involves customers building applications using vendor-supported programming languages and tools and then deploying them on cloud infrastructure. As in the SaaS model, the customer cannot manage or control the infrastructure – networks, servers, operating systems or storage systems – but has control over application deployment.

In the PaaS model, users must pay attention to application security, as well as issues related to API management, such as proof of access rights, authorization, and validation.

The PaaS model is secure, but the risk lies in insufficient system performance. The reason is that when exchanging data with PaaS providers, it is recommended to use encryption, which requires additional processing power. However, in any solution, the transfer of confidential user data must be carried out over an encrypted channel.

In the IaaS model, customers do not control anything. They only have control over operating systems, data storage, application deployment, and possibly limited control over the selection of network components.

This model has several built-in security capabilities without protecting the infrastructure itself. This means that users must manage and secure operating systems, applications, and content, typically through APIs.

If this is translated into the language of protection methods, then the provider must ensure:

- reliable access control to the infrastructure itself;
- fault tolerance of the infrastructure.

At the same time, the cloud client takes on many more protection functions:

- protection against network intrusions;
- protection of operating systems and databases (access control, protection against vulnerabilities, control of security settings);
- protection of end applications (antivirus protection, access control).

Thus, most of the protection measures fall on the shoulders of the consumer. The provider can provide typical protection recommendations or ready-made solutions, which will simplify the task for end users. The separation of responsibility for ensuring security between the client and the service provider is shown in the table. 4.1.

Table 4.1 - Separation of responsibilities

	Server	IaaS	PaaS	SaaS
Addition	Client	Client	Client	Provider
Data	Client	Client	Client	Provider
Execution environment	Client	Client	Provider	Provider
Connecting software	Client	Client	Provider	Provider
Operating System	Client	Client	Provider	Provider
Virtualization	Client	Provider	Provider	Provider
Servers	Client	Provider	Provider	Provider
Data repositories	Client	Provider	Provider	Provider
Network equipment	Client	Provider	Provider	Provider

4.3 Investigating of information protection methods in cloud access channels

4.3.1 Use of VPN technology for cloud services

The goal of Virtual Private Network (VPN) technologies is to isolate the data flows of one enterprise from the data flows of all other public network users to the maximum degree [8].

Since the main task of a VPN is to protect traffic, the virtual network must meet a large number of requirements and, first of all, have protection due to reliable cryptography, which guarantees protection against eavesdropping and changes. In addition, the VPN must have a reliable key management system and a crypto interface that allows crypto operations, for example, the "protected mail" operation, programs for encrypting disks, files, etc.

The variant, when the endpoints of the protected tunnel coincide with the endpoints of the protected message flow, is better from the point of view of security. In this case, full security of the channel is ensured along the entire path of message packets. In fig. 4.2 presents a protected virtual channel.

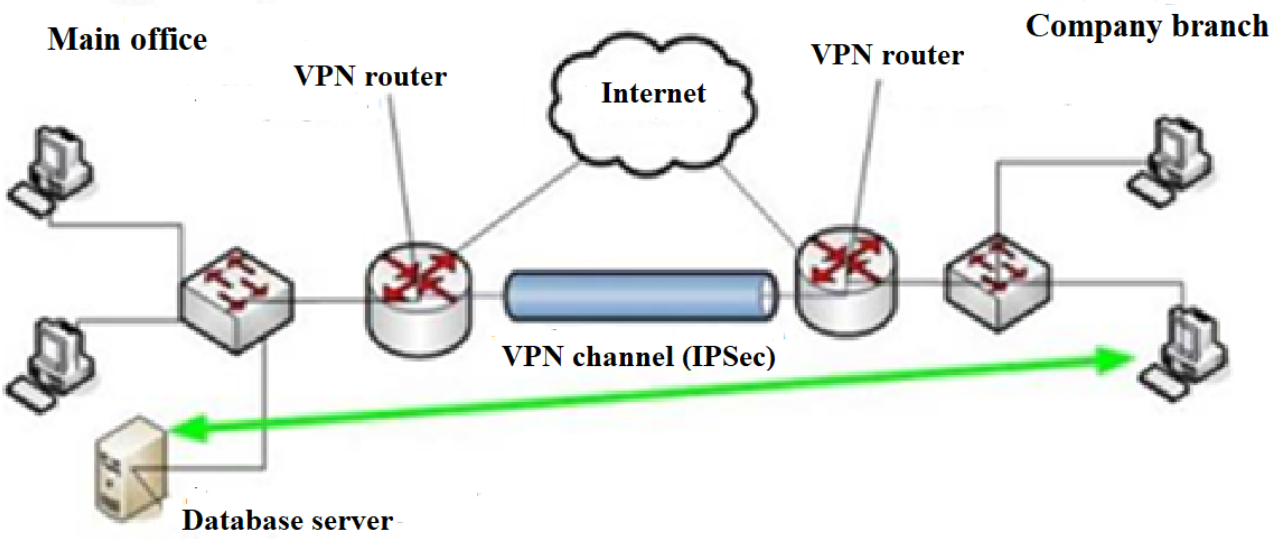

Figure 4.2 – Secure virtual channel

The creation of a secure tunnel is performed by the virtual network components operating on the nodes between which the tunnel is formed. These components are called tunnel initiator and terminator. The tunnel initiator encapsulates the packets into a new packet containing, along with the original data, a new header with information about the sender and recipient. Although all tunneled packets are IP packets, encapsulated packets can belong to any protocol type.

The route between the initiator and the terminator of the tunnel is determined by the normal routing of the IP network, which may be different from the Internet.

The reverse encapsulation process removes the new headers and directs each outgoing packet to the local protocol stack or destination on the local network. Encapsulation itself does not affect the security of message packets transmitted through the VPN tunnel. But thanks to encapsulation, the possibility of full cryptographic protection appears. The confidentiality of the encapsulating packets is ensured by their cryptographic closure, that is, they are encrypted, and the integrity and authenticity of the print is ensured by the formation of a digital signature.

The classification of VPNs by type of medium used is given below.

1) Secure VPN networks are the most common variant of private networks. With its help, it is possible to create a reliable and secure subnet on the basis of an unreliable network, usually the Internet.

2) Trust VPN networks are used in cases where the transmission environment can be considered reliable and only the task of creating a virtual subnet within a large network needs to be solved. Security issues become irrelevant. Examples of such VPN solutions are MPLS and L2TP. These protocols delegate the task of providing security to other protocols.

VPN is classified by the method of implementation.

1) VPN network in the form of special software and hardware. Implementation of the VPN network is carried out with the help of a special set of software and hardware tools. This implementation provides high performance and, as a rule, a high degree of security.

2) VPN network in the form of a software solution. They use a personal computer with special software that provides VPN functionality.

3) VPN network with an integrated solution. VPN functionality provides a set of tools, of which network traffic filtering is crucial.

According to the architecture of the technical solution, it is customary to distinguish three main types of virtual private networks:

- internal corporate VPN (Internet VPN);
- VPN with remote access (Remote Access VPN);
- inter-corporate VPN (Extranet VPN).

Internet VPN is used to connect several distributed branches of the same organization, which exchange data via open communication channels, into a single secure network. When organizing such a connection scheme, the presence of VPN servers, which is equal to the number of offices, is required.

This method is advisable to use both for regular branches and for mobile offices, which will have access to the resources of the "mother" company, as well as exchange data among themselves without problems.

An intranet is built on the same concepts and technologies that are used for the Internet, such as the client-server architecture and the Internet protocol stack (TCP/IP). All of the well-known Internet protocols, such as HTTP (web services), e-mail (SMTP), and file transfer (FTP), can be found in an intranet. Internet technologies are often used

to provide modern interfaces to the functions of information systems that host corporate data.

Advantages of Internet VPN:

- use of powerful cryptographic data encryption protocols to protect confidential information;

- reliability of functioning when performing such critical applications as automated sales systems and database management systems;

- the flexibility of managing the effective placement of a rapidly growing number of new users, new offices and new software applications.

In fig. 4.3 shows the Internet VPN connection scheme.

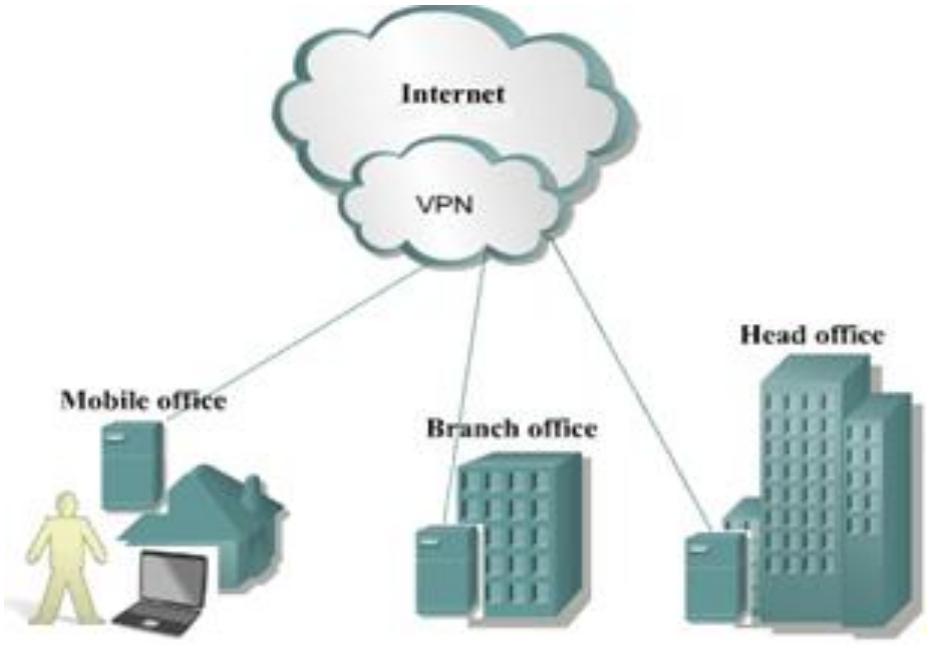

Figure 4.3 – Internet VPN connection diagram

Remote Access VPN is used to create a secure channel between a segment of the corporate network (central office or branch) and a single user who, working at home, connects to corporate resources from a home computer or, while on a business trip, connects to corporate resources using a laptop or smartphone, as schematically shown in fig. 4.4.

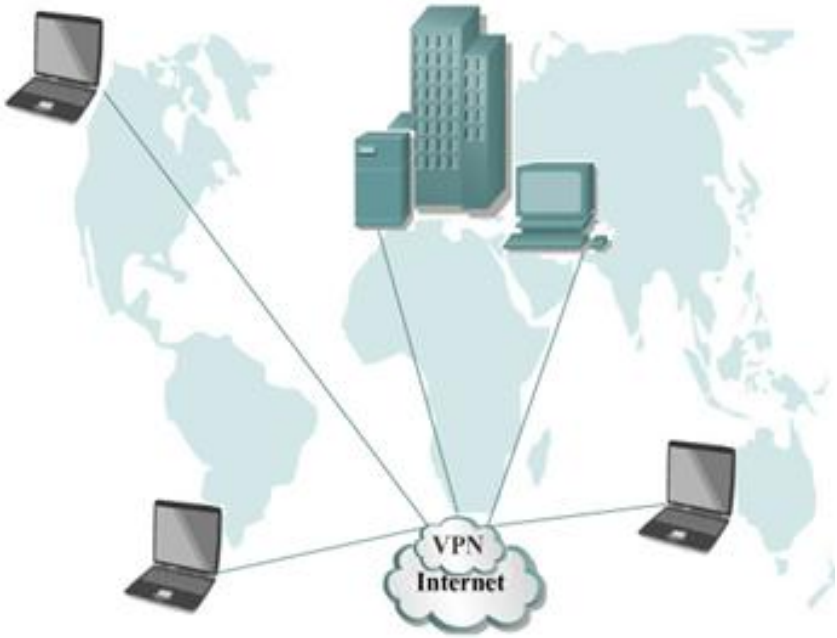

Figure 4.4 – VPN remote access scheme

Advantages of switching from privately managed dial networks to Remote Access VPN:

- the possibility of using local dial-in numbers instead of long-distance numbers allows you to significantly reduce costs for long-distance telecommunications;

- an effective system for establishing the authenticity of remote and mobile users ensures a reliable authentication procedure;

- high scalability and ease of deployment for new users added to it;

- focusing the company's attention on the main corporate business goals instead of being distracted by the problems of ensuring the operation of the network.

Extranet VPN is used for networks to which "external" users (for example, customers or clients) connect. The level of trust in them is much lower than in the company's employees, so it is necessary to provide special "borders" of protection that prevent or limit the latter's access to particularly valuable, confidential information. Extranet VPN is characterized by the use of standardized VPN products that guarantee interoperability with various VPN solutions that business partners might use in their networks.

VPNs are classified by the type of technical implementation. VPN based on firewalls. Firewalls of most manufacturers support tunneling and data encryption. All such products are based on the fact that the traffic passing through the firewall is encrypted. The encryption module shown in fig. is added to the software of the firewall itself. 4.5.

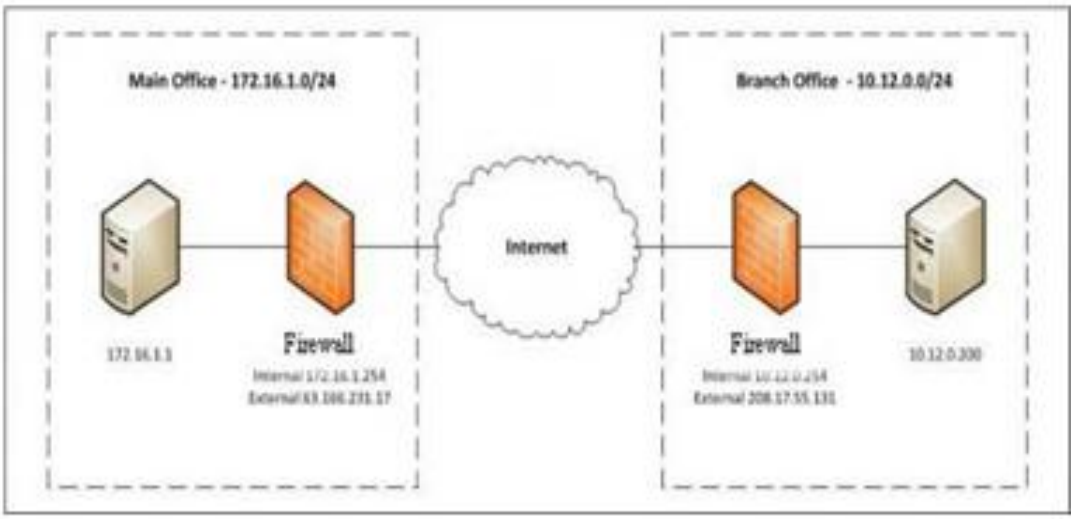

Figure 4.5 – VPN construction scheme based on firewalls

VPN based on routers. Another way to build a VPN is to use routers in Fig. to create secure channels. 4.6. Because all the information that comes out of the local network passes through the router, it is advisable to put the task of encryption on this router. Due to the fact that the router passes through itself all packets transmitted from the local network, it can also be used to encrypt these packets. In addition, the router can perform the function of decrypting incoming traffic.

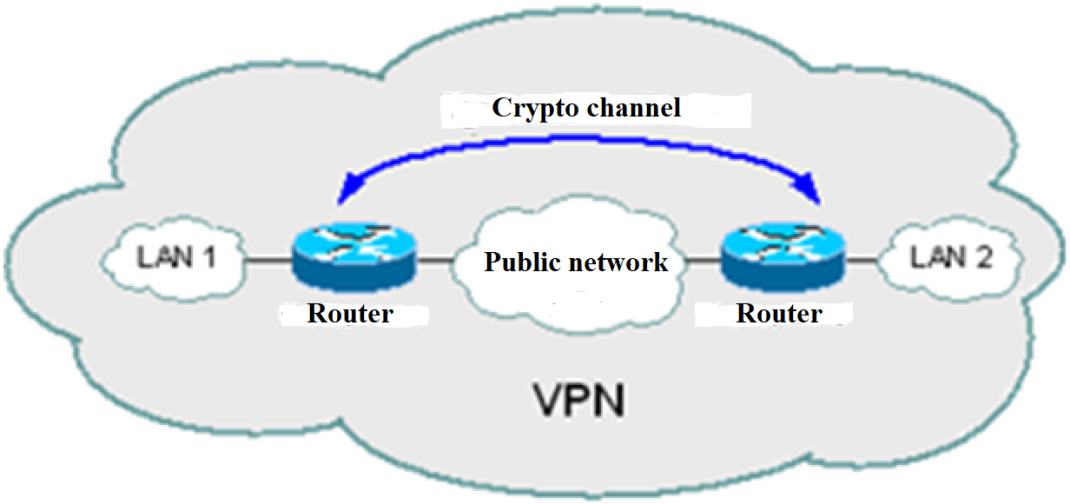

Figure 4.6 – VPN based on routers

A software-based VPN is the next approach to building a VPN and is a purely software solution. When implementing such a solution, specialized software is used, which runs on a dedicated computer and in most cases acts as a proxy server. A computer with such software can be located behind a firewall. In fig. 4.7 presents a hard-

ware-based VPN. The option of building a VPN on special devices can be used in networks that require high performance.

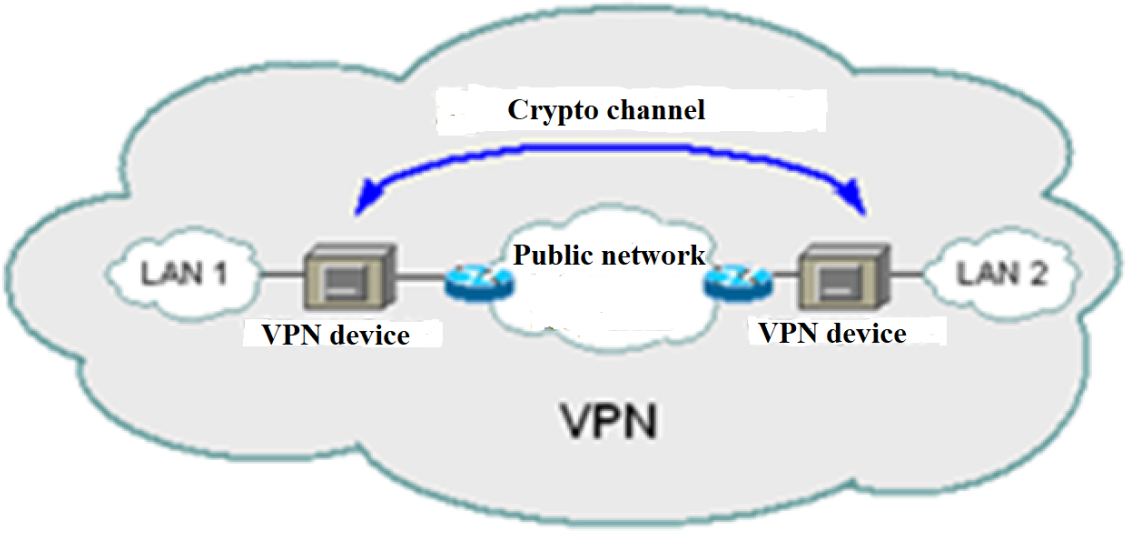

Figure 4.7 – Hardware-based VPN

VPNs can be classified by the type of medium used as follows.

1) Secure VPN networks are the most common variant of private networks. With its help, it is possible to create a reliable and secure subnet on the basis of an unreliable network, usually the Internet. Examples of secure VPNs are IPSec, OpenVPN, and PPTP.

2) Trust VPN networks are used in cases where the transmission environment can be considered reliable and only the task of creating a virtual subnet within a larger network needs to be solved. Security issues become irrelevant. Examples of such VPNs are: MPLS and L2TP. It is more correct to say that these protocols transfer the task of ensuring security to others, for example L2TP and, as a rule, are used in conjunction with IPSec.

4.3.2 Protecting information using a firewall

An inter-network screen is a locally (one-component) or functionally distributed means (complex) that implements control of information that enters the automated system or leaves the automated system. It provides protection of the automated system by filtering information, that is, analyzing it according to a set of criteria and making a decision on its distribution to or from the automated system (Fig. 4.8).

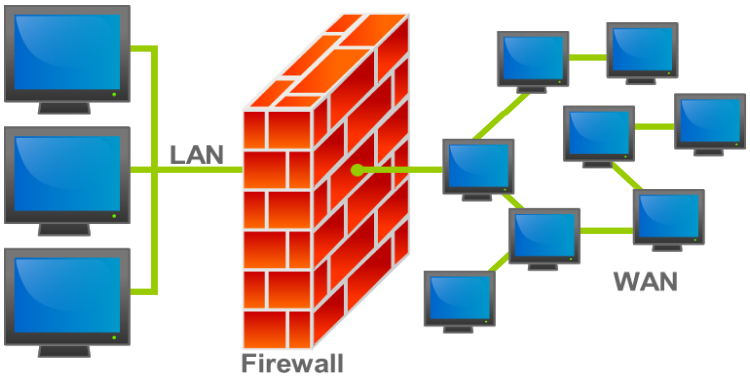

Figure 4.8 - An illustration of a network-based firewall within a network

A firewall should be used to increase computer security by limiting information coming from other computers, controlling any unauthorized actions of applications and programs that try to access the information resources of the local computer.

Organizationally, the firewall is part of the protected network.

The firewall is not symmetrical. Parameters limiting access from the internal network to the external network and vice versa are set separately for it. The firewall must take into account the information exchange protocols that form the basis of the internal and external network.

Firewalls manage the network traffic that passes inside the local network and allow only authorized traffic to pass through the network connection, controlling the network interaction between computers on the global and local networks. Firewalls allow you to mask the IP addresses of hosts in the middle of a local network using an operation called Network Address Translation (NAT).

Masking an IP address makes it invisible to external users who, for example, send e-mail messages to an internal user by routing them to a mail gateway that forwards them to their destination.

Firewalls allow you to control the access of network users to various network services. This task is solved by configuring the firewall, in which you can allow or block access to a particular local network service using Access Control Lists (ACLs). ACLs provide flexible access control capabilities. With their help, you can decide access to certain services and deny access to all other services, or, conversely, block access to certain services and allow access to all other services. Well-configured firewalls not only block unauthorized requests from external computers, but also try to identify the authors of the request along with the immediate notification of the user to the system administrator about the attempts of such requests.

Firewalls consist of a set of software and hardware components, which include the following.

1) Bastion host. This host is a computer with a protected version of the OS connected to a local and wide area network. All other firewall components and necessary services, such as Telnet, DNS, FTP, SNMP, as well as user authentication tools, are installed on the bastions in the computer.

2) Router with packet filtering. A regular router that forwards IP packets to the specified address. A packet filtering router performs an additional function of inspecting incoming IP packets. A router with packet filtering is sometimes called a secure router. Secure routers do not inspect packet content, but only deal with packet headers, monitoring source and destination IP addresses, protocols used, services, ports, and other information specified in the ACL.

3) Application gateways (application gateways). They are used on the Bastion host and limit connections to certain applications. For this purpose, intermediary services are used, which are installed on the gateway separately for each application that is allowed network interaction through the firewall. Only those network services for which broker services are installed can receive and send network traffic through application gateways, and broker services can be configured to allow access only to a specific, limited set of application facilities. Thus, application gateways greatly enhance the ability to create such a security policy that ensures authentication of network users and registration logging. An example of an application-level gateway is a proxy server that manages outgoing network traffic and authenticates users.

Firewalls perform packet filtering by inspecting the headers of incoming packets against certain criteria established by packet filtering rules. Packets coming from both inside and outside the local network are filtered, and the filter works asymmetrically, processing incoming and outgoing packets differently. So. different filtering rules should be used to filter incoming and outgoing packets.

When a packet arrives at the firewall, which includes a router with packet filtering, it extracts headers from the packet and performs syntactic analysis and header verification. At the same time, only headers related to TCP, IP, UDP protocols are checked. Next, the packet filtering rules are sequentially applied to the packet, and in the order in which they are stored in the firewall's ACL list.

The rules are applied taking into account the following principles:

- if during ACL review a rule is found that allows the packet to pass, it is immediately forwarded to its destination;

- if a rule is found that prohibits the passage of the packet, it is immediately rejected;

- if when viewing the ACL, it turns out that there are no rules for the package that allow it th pass, the packet is automatically discarded.

To create a packet filtering rule, you should specify: the action performed when the rule criteria match the packet parameters; packet processing protocol and port number for receiving the packet. Incorrect order of writing rules can lead to complete blocking, connection or rejection of correct packets.

Packet filtering is an effective means of protecting various services from network attacks, but packet filtering techniques are ineffective against attacks that do not depend on network services. For example, from packet IP spoofing attacks that can be applied to any network service. This type of attack is called IP spoofing. To carry out such an attack, a hacker from an external host replaces the source real IP address of a packet with a fake one that allows packets to pass through. This fake IP address may be the IP address of an internal network host. If the firewall is not configured properly, a packet with a spoofed IP address can be allowed into the network.

Another example of a network attack against which packet filtering is powerless is bypassing the network protection system with routing information specified in the transmitted packet. If the firewall is not configured to drop packets containing routing information, such an attack can succeed.

Packet-filtering firewalls can miss a packet-fragmentation attack, in which an attacker divides forwarded packets into small pieces and sends them to a packet-filtering router. In a fragmented packet, the port number of the target host must contain only the very first fragment, and the rest of the fragments contain only the message itself, so if the first packet is missed, the firewall will also miss the others.

Another disadvantage of packet-filtering routers is that they lack packet content inspection, making them unsuitable for protection against data-driven attacks.

Application gateways allow for stricter security policy rules than packet filtering routers.

Application gateways use special programs called broker services to manage traffic between WAN and LAN hosts. To protect applications, you need to install a

separate service, without which the application will not be able to provide its services to network users.

When using application gateways, authorized users can access intermediary services to obtain the service they need, but they are not allowed to access the application gateway because it poses a security threat to the firewall.

Unlike packet filtering gateways, application gateways prohibit the direct exchange of packets between internal and external hosts. To resolve the issue of providing access to the intermediary service, the bastion host may perform additional user authentication. For this, the technology of one-time passwords generated by a cryptographic device can be used. To strengthen protection, such means of authentication can be implemented separately for each of the intermediary services.

The main advantage of application gateways is that they allow you to strictly limit access to all programs and services used in the local network, both external and internal hosts.

The disadvantage is the limitation of users' freedom of action, as well as the need to install additional software tools on each host.

Technologies for creating stateful channel gateways are used to correct the mentioned shortcomings of application gateways.

Channel gateways directly connect the TCP / IP ports of the bastion host to the network host and do not check the network traffic that passes and this allows to increase the speed of the firewall.

The firewall provides protection only around the perimeter of the computer system. It is not useful against attacks from the middle of the network, particularly against data-driven attacks, in which externally secure data is transmitted to the local network and then used to attack the network from the middle.

A firewall is equipped with a means of checking incoming e-mail for viruses, but it is generally not able to reliably protect the system from virus intrusion.

If you have several firewalls installed on your computer, you should not turn them on at the same time, as a result of turning them on at the same time, some programs will stop working correctly.

Installation of a firewall provides effective protection only through its precise adjustment, performed over a long period of system operation.

In order for the firewall to perform the functions it is required for comprehensive network protection, it is necessary that all packets coming through the network and

going into the network pass through it. If this rule is not followed, or partially followed, then all actions aimed at creating a secure server using a firewall will be useless.

As a result of consideration of different types of network screens, it is possible to distinguish 5 main types of such screens.

Five types of firewall include the following.

1. Packet filtering firewall.
2. Circuit-level gateway.
3. Application-level gateway (proxy firewall).
4. Stateful inspection firewall.
5. Next-generation firewall (NGFW).

The advantages and disadvantages of different types of firewalls are listed in Table 4.2.

Choosing the right type of firewall means answering questions about what the firewall is protecting, which resources the organization can afford and how the infrastructure is architected. The best firewall for one organization may not be a good fit for another.

Issues to consider include the following:

- What are the technical objectives for the firewall? Can a simpler product work better than a firewall with more features and capabilities that may not be necessary?
- How does the firewall itself fit into the organization's architecture? Consider whether the firewall is intended to protect a low-visibility service exposed on the internet or a web application.
- What kinds of traffic inspection are necessary? Some applications may require monitoring all packet contents, while others can simply sort packets based on source/destination addresses and ports.

Many firewall implementations incorporate features of different types of firewalls, so choosing a type of firewall is rarely a matter of finding one that fits neatly into any particular category. For example, an NGFW may incorporate new features, along with some of those from packet filtering firewalls, application-level gateways or stateful inspection firewalls.

Whichever type(s) of firewalls you choose, keep in mind that a misconfigured firewall can, in some ways, be worse than no firewall at all because it lends the dangerously false impression of security while providing little to no protection.

Table 4.2 - Advantages and disadvantages of different types of firewalls

FIREWALL TYPE	ADVANTAGES	DISADVANTAGES
Packet filtering firewall	■ A single device can filter traffic for the entire network ■ Efficient and fast at processing packets ■ Enables complex security policies through filtering on protocol headers ■ Inexpensive ■ Minimal impact on other resources, network performance, end-user experience	■ Incapable of filtering at the application layer ■ Lacks broad context of other firewall options ■ Can be difficult to securely configure ■ Lacks features like user authentication, logging ■ Vulnerable to spoofing attacks ■ Access controls lists can be difficult to set up and manage
Circuit-level gateway	■ Provides privacy for data passing in/out of private network ■ More efficient processing traffic than application-level gateways ■ Relatively inexpensive ■ Easier to set up and manage ■ Minimal impact on end-user experience	■ Protects circuits (network sessions) rather than individual packets ■ Requires modification to network protocol stack ■ Incapable of content filtering ■ Should be used in conjunction with other firewall technologies ■ Does not offer application-layer monitoring
Application-level gateway	■ Capable of detecting and blocking attacks not visible at the OSI model network or transport layers ■ Obscures private network details ■ Protects user anonymity ■ Enables more fine-grained security controls	■ Complex to configure and maintain ■ High processing overhead ■ Requires a proxy be set up for every network application in use ■ Can affect network performance
Stateful inspection firewall	■ Capable of blocking types of attacks that exploit protocol vulnerabilities ■ Can operate with fewer open ports, reducing attack surface ■ Capable of blocking many types of denial-of-service attacks	■ Can require high degree of skill to securely configure ■ Does not support authenticated connections ■ Not effective against exploits of stateless protocols ■ High processing overhead
Next-generation firewall	■ Provides traditional firewall functionality combined with other security functions, including intrusion detection/prevention systems (IDS/IPS), advanced threat intelligence, malware scanning and others ■ Capable of monitoring network protocols from the data link layer (Layer 2 of the OSI model) through the application layer (Layer 7 of the OSI model) ■ Offers substantive logging capabilities ■ Can be more efficient at processing network traffic than combination of firewall plus IDS/IPS and malware scanning	■ Consolidation of security functions makes the NGFW a single point of failure ■ Requires high front-end investment of resources to acquire, configure and deploy these complex systems ■ Depending on architecture, may be processing-intensive ■ Not all organizations will require all the functionality of an NGFW ■ Can hinder network performance ■ More expensive than other firewall options

4.4 Using IDS/IPS and choosing systems to protect cloud services

The principles of operation and purpose of IDS/IPS systems are shown in fig. 4.9.

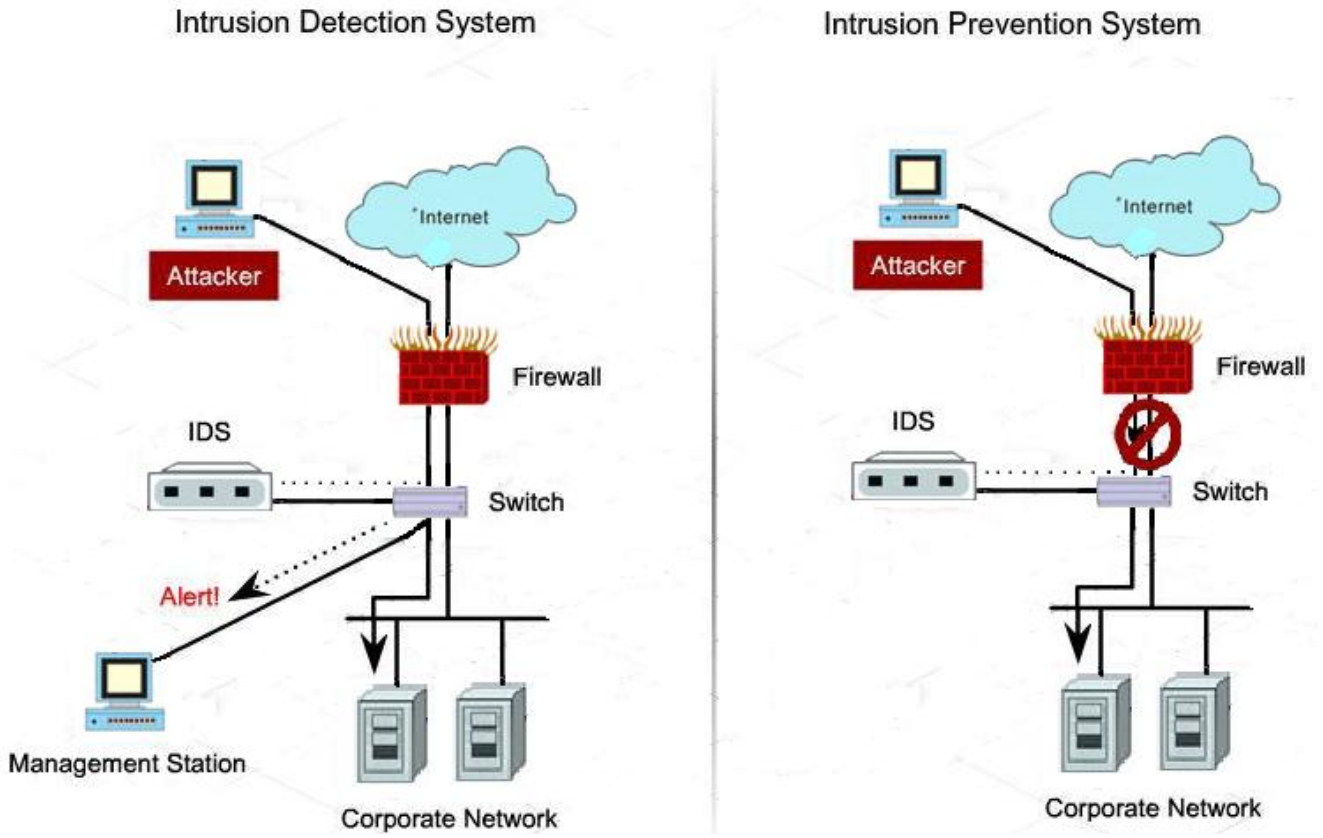

Figure 4.9 - Principles of operation and purpose of IDS/IPS systems

The task of the Intrusion Detection System (IDS) is to detect and register attacks, as well as to notify when a certain rule is triggered [10]. Depending on the type, IDS can detect various types of network attacks, detect unauthorized access attempts or elevation of privileges, the appearance of malicious software, monitor the opening of a new port, etc. Unlike a firewall, which monitors only session parameters (IP, port number and state of connections), IDS "peers" into the middle of the packet (up to the seventh level of OSI), analyzing the transmitted data. There are several types of intrusion detection systems. Application protocol-based IDS (APIDS), which monitor a limited list of application protocols for specific attacks, are quite popular. Typical representatives of this class are PHPIDS, which analyzes requests to PHP applications, Mod_Security, which protects the web server (Apache), and GreenSQL-FW, which blocks dangerous SQL commands.

Network Network Intrusion Detection System (NIDS) is more universal, which is achieved thanks to Deep Packet Inspection (DPI) technology, deep packet inspection. They control not one specific application, but all traffic, starting from the channel level.

For some packet filters, the ability to "look inside" and block the danger is also implemented. As an example, we can cite the OpenDPI and Fwsnort projects. The latter is a program for converting the Snort signature base into equivalent blocking rules for «iptables». But initially the firewall is sharpened for other tasks, and the DPI technology is "overhead" for the engine, so the functions for processing additional data are limited to blocking or marking strictly defined protocols. IDS just marks (alert) all suspicious activities. To block the attacking host, the administrator manually reconfigures the firewall while viewing the statistics. Naturally, we are not talking about any real-time reaction here. That is why Intrusion Prevention System (IPS) - attack prevention systems are more interesting today. They are based on IDS and can independently rebuild the packet filter or terminate the session by sending a TCP RST. Depending on the principle of operation, IPS can be installed "in a gap" or use traffic mirroring (SPAN) received from several sensors. For example, a Hogwash Light BR, which operates at the second OSI layer, is installed in the gap. Such a system may not have an IP address, which means it remains invisible to a hacker.

In ordinary life, the door is not only locked, but also additionally protected by leaving a guard near, because only in this case you can be sure of safety. IPS systems act as such defenders in IT. They are often confused with antiviruses that have a proactive protection module. But IPS, as a rule, do not use signatures, which means that they do not require constant updating of databases. They control many more system parameters: processes, integrity of system files and registry, log entries and much more.

In order to fully own the situation, it is necessary to monitor and map events at both the network level and the host level. Hybrid IDS were created for this purpose. Such systems are often referred to as Security Information Management (SIM). Among the OpenSource projects, Prelude Hybrid IDS is interesting, which collects data from almost all OpenSource IDS/IPS and understands the log format of various applications. Support for this system was discontinued a few years ago, but compiled packages can still be found in Linux and BSD repositories.

The modern Internet carries a huge number of threats, so highly specialized systems are no longer relevant. It is necessary to use a comprehensive multifunctional solution that includes all components of protection: firewall, IDS / IPS, antivirus, proxy

server, content filter and antispam filter. Such devices were named Unified Threat Management (UTM), unified control of threats. Examples of UTM include Trend Micro Deep Security, Kerio Control, Sonicwall Network Security, FortiGate Network Security Platforms and Appliances.

Suricata is a free, open source, mature, fast and reliable network threat detection engine capable of detecting intrusions in real time. In IDS, built-in IPS intrusion prevention, NSM network security monitoring and offline processing of captured packets. Suricata is rapidly developing, focused on security, ease of use and efficiency. Suricata inspects network traffic using powerful and extensive rules and signature language, and has powerful scripting support to detect complex threats.

Some modern IDSs with a long history, including Snort, do not use multi-processor/multi-core systems very efficiently, leading to problems with processing large volumes of data. Suricata works in multi-threaded mode.

In general, the modular layout allows you to quickly connect the necessary element for capturing, decoding, analyzing or processing packets. Blocking is carried out using the OS's standard packet filter. Engine automatically detects and parses protocols (IP, TCP, UDP, ICMP, HTTP, TLS, FTP, SMB, SMTP and SCTP), so it is not necessary to bind to the port number in the rules (as Snort does), it is enough to set the action for the desired protocol.

Actually, all interfaces and analyzers written for Snort (Barnyard, Snortsnarf, Sguil, etc.) work with Suricata without modifications. This is also a big plus. HTTP exchange is detailed in the Apache standard format file.

Rules are the basis of the detection mechanism in Suricata. Here, the developers did not invent anything yet, but allowed to connect rulesets created for other projects. In the first releases, the support was only partial, and the engine did not recognize and load some rules, but now this problem is solved. A custom rules format has also been implemented, which outwardly resembles Snort's. A rule consists of three components: an action (pass, drop, reject, or alert), a header (source and destination IP/port), and a description of what to look for. Variables (flowint mechanism) are used in the settings, which allow, for example, to create counters. At the same time, information from the stream can be saved for later use. This approach, used for tracking password attempts, is more effective than the method used in Snort, which operates on the limit value of the trigger.

Samhain HIDS host-based intrusion detection system provides file integrity checking and log file monitoring / analysis, as well as rootkit detection, port monitoring, detection of rogue SUID executables and hidden processes. Samhain was designed to monitor multiple hosts with potentially different operating systems, providing centralized logging and maintenance, although it can also be used as a standalone application on a single host.

Samhain is an open source application for POSIX systems (Unix, Linux, Cygwin / Windows), protected by an OpenSource license.

Samhain refers to a host IDS that protects an individual computer. It uses several analysis methods that allow you to fully cover all the events that occur in the system:

- creation of a database of signatures of important files at the first launch and its subsequent comparison with the "live" system;
- monitoring and analysis of journal entries;
- login/logout control;
- monitoring of connections to open network ports;
- control of files with set SUID and hidden processes.

The program can be run in invisible mode (the kernel module is used), when kernel processes cannot be detected in memory. Samhain also supports monitoring multiple nodes running different OSes, with all events logged in one place. At the same time, agents installed on remote nodes send all collected information (TCP, AES, signature) via an encrypted channel to the server (yule), which stores it in a database (MySQL, PostgreSQL, Oracle). In addition, the server is responsible for checking the status of client systems, distributing updates and configuration files. Several options for notifications and sending collected information have been implemented: e-mail (mail is signed to avoid forgery), syslog, log file (signed), Nagios, console, etc. Management can be done by multiple administrators with clearly defined roles.

Stonesoft's next-generation IPS provides flexibility and scalability for data center and internal network deployments. It was designed for real-time threat protection and application control, providing superior visibility and situational awareness for enterprise network data.

Using multi-level traffic normalization and inspection technology, Stonesoft IPS protects against Advanced Evasion Techniques (AET) without compromising the speed or availability of network traffic. Stonesoft SSL VPN provides transparent access for all

applications with flexible single sign-on. If you need to use a mix of public and private cloud services, deploy federated authentication to separate authentication from access.

This solution was developed by a Finnish company that produces enterprise-class products in the field of network security. It implements all the necessary functions: IPS, protection against DDoS and 0day attacks, web filtering, support for encrypted traffic, etc. With the help of StoneGate IPS, you can block virus, spyware, certain programs (P2P, IM and others). For web filtering, a permanent database of sites, which is updated and divided into several categories, is used. Special attention is paid to protection against circumvention of AET security systems. Transparent Access Control (TAC) technology allows dividing the corporate network into several virtual segments without changing the real topology and setting individual security policies for each segment. Traffic inspection policies are configured using templates containing typical rules. These policies are created offline. The administrator checks the created policies and downloads them to the remote IPS nodes. Similar events in StoneGate IPS are processed according to the principle used in SIM / SIEM systems, which significantly facilitates analysis. Several devices can be easily combined into a cluster and integrated with other StoneSoft solutions – StoneGate Firewall / VPN and StoneGate SSL VPN. At the same time, management is provided from a single management console (StoneGate Management Center). The console allows you not only to configure the operation of IPS and create new rules and policies, but also to monitor and view logs.

StoneGate IPS is delivered both as a hardware complex and as a VMware image. The latter is intended for installation on own equipment or in a virtual infrastructure. And by the way, unlike the creators of many similar solutions, the developer company lets you download a test version of the image.

IBM Security Network Intrusion Prevention System – an attack prevention system developed by IBM, uses a patented protocol analysis technology that provides preventive protection, including against 0-day threats. As with all products of the IBM Security series, its basis is a protocol analysis module - Protocol Analysis Module (PAM), which combines a traditional signature method of detecting attacks (Proventia OpenSignature) and a behavioral analyzer. At the same time, PAM distinguishes between 218 application protocols (attacks via VoIP, RPC, HTTP, etc.) and data formats such as DOC, XLS, PDF, ANI, JPG to predict where malicious code can be injected. More than 3000 algorithms are used for traffic analysis, 200 of them "catch" DoS attacks. Firewall functions allow access only to certain ports and IPs, eliminating the need

to involve an additional device. Virtual Patch technology blocks viruses at the stage of distribution and protects computers until an update is installed that eliminates a critical vulnerability. If necessary, the administrator can create and use the signature himself. The application control module allows you to manage P2P, IM, ActiveX elements, VPN tools, etc. And if necessary, block them. The implemented DLP module monitors attempts to transmit confidential information and move data in the protected network, which allows you to assess risks and block leakage. Eight data types are recognized by default. Currently, most vulnerabilities affect web applications, so the IBM product includes a special Web Application Security module that protects systems against common types of attacks: SQL injection, LDAP injection, XSS, JSON hijacking, PHP file-includes, CSRF, etc.

There are several options for actions when an attack is detected - blocking the host, sending a warning, recording the attack traffic (in a file compatible with tcpdump), moving the node to a user-defined quarantine, and some others. Policies are written down to each port, IP address, or VLAN zone. The High Availability mode ensures that in the event of failure of one of the several IPS devices available in the network, the traffic will go through the others, and the established connections will not be broken. All subsystems inside the device – RAID, power supply, cooling fan – are duplicated. If you have multiple devices, you usually get IBM Security SiteProtector, which provides centralized management, performs log analysis, and generates reports.

The choice in each specific case depends on the budget, network topology, and required protection functions. Commercial solutions receive support and are provided with certificates, which allows the use of these solutions in organizations that are engaged in the processing of personal data, including secondary distribution under the OpenSource license. Snort is well documented, has a fairly large base and a good track record.

CONCLUSION

The research in this thesis was motivated by the fact that IoT devices, despite their proliferation in domestic environments and CNIs, introduce tremendous security flaws, and subsequently are subject to a range of cyber attacks. As such devices are often deeply embedded in networks, IoT may be considered as being the ‘weakest link’ for breaking into a secure infrastructure. Consequently, there is a significant need for the development of novel mechanisms to improve not only the defense of IoT against a range of cyber attacks, but also the detection of such attacks, and subsequently their mitigation from IoT networks.

Due to their limitations in computational power and their heterogeneity, securing the IoT ecosystem is considered as being a great challenge. This is because it is not feasible for IoT devices with restricted computational power to execute computationally intensive and latency-sensitive security tasks. As a result, it is not possible to employ complex and robust security measures. Moreover, the heterogeneity which surrounds IoT devices in terms of their hardware, software, and protocols poses as an obstacle towards developing and deploying security mechanisms that can endure with the scale and range of devices.

It is possible to increase the efficiency of the protection system due to the multi-level architecture of the system. At each border of the protective barrier, the overall probability of information protection increases. The more protective barriers - the higher the level of protection.

A study was carried out of the dependence of the increase in the probability of threat reflection in relation to one-level protection and two-level protection. Suggested to use protection at the level Fog computing and at the Cloud computing.

On an example with the same probabilities equal to 0.5, an increase of 0.25 was calculated when using a two-stage instead of a single-stage protection, and when introducing three-stage protection, the increase was 0.125 compared to a two-stage.

Research has been done of information protection methods in cloud access channels. Suggested to use VPN technology and firewalls.

REFERENCE

1. Prahlada Rao B.B., Payal Saluja. Cloud computing for Internet of Things & sensing based applications. [Електронний ресурс]. – Режим доступу: https://www.researchgate.net/publication/261422509

2. Adnan Khalid, Muhammad Shahbaz. Adaptive Deadline-aware Scheme (ADAS) for Data Migration between Cloud and Fog Layers. [Електронний ресурс]. – Режим доступу: https://www.researchgate.net/publication/324440408

3. Isaac Odun-Ayo, Chinonso Okereke, Orovwode hope Evwieroghene. Cloud Computing and Internet of Things - Issues and Developments. [Електронний ресурс]. – Режим доступу: https://www.researchgate.net/publication/333402518

4. Fahmida Naveed, Muhammad Rizwan, Aysha Shabbir, Maryam Shabbir and Fahad Ahmad. Cloud Computing Serving as a Solution to the IoT Generated Data // Bahria University Journal of Information & Communication Technologies, vol. 11, issue II, December 2018.

5. Mohammad Aazam, Imran Khan, Aymen Abdullah Alsaffar, Eui-Nam Huh. Cloud of Things: Integrating Internet of Things and cloud computing and the issues involved. [Електронний ресурс]. – Режим доступу: https://www.researchgate.net/publication/271462399

6. Evaluating Your IoT Security. How to approach the new threats and consequences facing your business with the Internet of Things. Microsoft. [Електронний ресурс]. – Режим доступу: www.InternetofYourThings.com

7. Massimiliano Rak, Massimo Ficco and Ermanno Battista, Valentina Casola, Nicola Mazzocca. Developing secure cloud applications. [Електронний ресурс]. – Режим доступу: https://www.researchgate.net/publication/275640297

8. Mohammad Aazam, Pham Phuoc Hung, Eui-Nam Huh. Smart Gateway Based Communication for Cloud of Things. [Електронний ресурс]. – Режим доступу: https://www.researchgate.net/publication/269303095

9. Хмарні технології. Основні поняття і типи хмарних сервісів [Електронний ресурс]. – Режим доступу: https://profit.kz/articles/10305/Oblachnie-tehnologii-

10. Система запобігання вторгнень. [Електронний ресурс]. – Режим доступу: http://www.stonesoft-security.co.uk/product/stonesoft-intrusion-prevention-system/.

11. Kenneth Hui. Data Encryption in the Cloud. Part 4: AWS, Azure and Google Cloud. [Електронний ресурс]. – Режим доступу: https://cloudarchitectmusings.com/2018/03/09/data-encryption-in-the-cloud-part-4-aws-azure-and-google-cloud/