На базі даної платформи Arduino NANO можна розширити функціонал пристрою підключенням модулів виведення поточного часу, температури, інтенсивності освітлення і відповідно керуванням освітлювальними приладами, приладами обігріву, включення електроприладів за заданим часом або на заданий термін тощо.

Список використаних джерел

1. Google Play. Додатки [Електронний ресурс]. – Режим доступу: https://play.google.com/store/apps/details?id=com.beragumbo.GyverMatrixBT
2. MDFLY electronics New products. [Електронний ресурс]. – Режим доступу: http://www.mdfly.com/
3. Arduino контролери [Електронний ресурс]. – Режим доступу:https://arduino.ua/prod166-arduino-nano-v3-0-avr-atmega328-p-20au-s-kabelem-mini-usb-i-raspayannimi-razyomami
4. Трехцветные светодиоды RGB [Електронний ресурс]. – Режим доступу: http://ledno.ru/svetodiody/trexcvetnye-rgb.html
5. Усе про світло діоди [Електронний ресурс]. – Режим доступу:http://elektrotovary.te.ua/index.php?route=information/news&news_id=6
6. Светодиодная RGB матрица [Електронний ресурс]. – Режим доступу:http://arduino.ua/prod340-svetodiodnaya-matrica-8h8-rgb-60mm

---

УДК 004.056.55

THE PRINCIPLES OF MODERN MESSENGERS WORK FOR TRANSFER ENCRYPTED DATA

D. Shuliachenko, T. Chornyi, students of group MCEs-171
Scientific supervisors: **S.V. Zaitsev,** Doctor of Technical Sciences, assistant professor
Chernihiv National University of Technology

There is a high competition in the area of messengers nowadays. The access to the Internet is not only on PC but also on smartphones. This fact allowed messengers to become the most popular app.

Each messenger has their audience which agitates to use exactly its service. After all users should register lots of accounts in different services and download lots of applications to have the opportunity for communication with all necessary people immediately.

The Internet which we know now exists due to open standards. All layers of network interaction, starting from the physical layer (transmission and reception of raw bit streams over a physical medium) to the application layer (HTTP, E-mail) are opened and accessible for anybody. Anybody can make his or her own web site, browser, e-mail client. But it is not protected.

That is why we have many operation systems, which can work with the Internet, and we have the variety of devices and applications that support popular protocols.

The question of building information technology for transmission encrypted data in modern mobile systems was investigated by foreign and domestic scientists: B. Holdshtein, A. Pinchuk, A. Sukhovytskyi, P. Sermos, A. Takanen, V. Polinovskyi, M. Houh.

There is a problem of creating distributed client-server systems with cryptographic protection in the application layer, which can support loads of users and provide appropriate protection of information.

The modern stage of data transmission development system in real-time and its high heterogeneity require from network equipment clear interaction and the ability to guarantee the qualitative data transmission from one network segment to another. One of the most perspective area in the development of such systems nowadays is creation of server base on IP protocol, which allows to create the real-time data transmission systems with agreement of requirement of information confidentiality. There are loads of information technologies, which are made for this aim. However, many of them cannot provide the appropriate level of data transmission protection with high quality of communications.

Therefore, the research started from the analysis of existing technologies for REST-server creating. This server provide encrypted messages and files transmission. Also, there was provided the analysis of databases for storing main information of the system.

During the work were offered new ways of server module services creation. They are based on the platform with open code Node.js. This gives us such advantages:

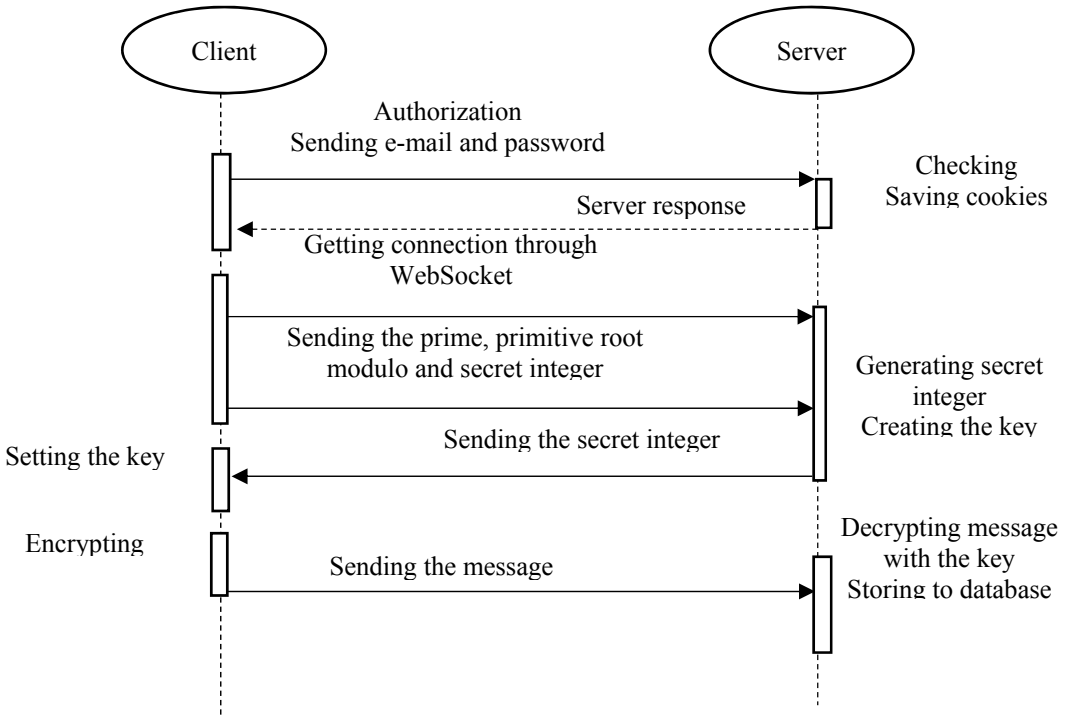
- Asynchronous singlethreading model of requests execution;
- Non-blocking input/output;
- Modules system CommonJS;
- Engine JavaScript Google V8.

Also, there were provided servers utilities for encrypted data transmission, Web client services for encrypted data transmission and visualization; developed new REST-server model due to using of asynchronous server platform NodeJs and NoSQL database MongoDB.

Individual modules were created by authors. These modules execute functionality of REST-server on the platform with an open code NodeJs for transmission and saving encrypted data. Server software was written on JavaScript. HTTPS and WebSocket protocols are used in services for transmission the data. Services use symmetric algorithm block ciphers Advanced Encryption Standard (AES) for encrypting messages with 256 bits

key length. In addition, asymmetric cryptographic HTTPS protocol is used for sending the key. Authenification module uses HTTPS protocol and 128 bits algorithm of hashing MD5.

The sequence of user actions for connecting to the server (Picture 1.1), the client sends his login and password to the server, after that the server creates an md5 cache and compares it with the cache which is stored in the database of the corresponding user. In case of validity, the user sends information to the server, and the client sends an identifier, generator and a private number module for generating a shared data encryption key. In turn, the server creates a key on the database and passes its private number, after which the client creates the corresponding key. Next, the client sends requests for information about users and groups. When a user clicks on a group or another user, the client requests a mailing history, by default it is the last 50 messages. When a message is sent to another user, the messages are encrypted on the client side, and transmitted to the server where it is decrypted, after that it is again encrypted but the database key, and it is stored in the database, and then sent to the user or group if they are online. If the client is offline, then the appropriate marks are created in the database.



Picture 1 - Sequence diagram during sending message

References

1. Information about platform NodeJS [Electronic resource]. – Access: URL: https://nodejs.org/uk/
2. Information about MongoDB database [Electronic resource]. – Access: URL: https://docs.mongodb.com
3. Information about protocol HTTPS [Electronic resource]. – Access: URL: https://uk.wikipedia.org/wiki/HTTPS
4. Information about protocol WebSocket [Electronic resource]. – Access: URL: https://uk.wikipedia.org/wiki/WebSocket
5. Information about additional module Express [Electronic resource]. – Access: URL: http://expressjs.com/uk/
6. Information about construction technology of REST-server [Electronic resource]. – Access: URL: https://uk.wikipedia.org/wiki/REST
7. Information about Diffie–Hellman algorithm [Electronic resource]. – Access: URL: http://www.intuit.ru/studies/courses/691/547/lecture/12391
8. Information about AES algorithm [Electronic resource]. – Access: URL: http://bit.nmu.org.ua/ua/student/metod/cryptology
9. C. Sgaras Forensics acquisition and analysis of instant messaging and VoIP applications / C. Sgaras, M. Kechadi, A. Le-Khac // Lecture Notes in Computer Science. – Vol. 8915. – 2015. – P. 188-199.
10. Jyoti S. Whatsapp, Skype, Wickr, Viber, Twitter and Blog are Ready to Asymptote Globally from All Corners during Communications in Latest Fast Life / S. Jyoti // Research Journal of Science and Technology. – 2014. – Vol. 6. – P. 101-116.
11. P. Fiadino Online Social Networks anatomy: On the analysis of Facebook and WhatsApp in cellular networks / P. Fiadino, P. Casas, M. Schiavone.