Journal of Harbin Engineering University
ISSN: 1006-7043
Vol 44 No. 10
October 2023

An Application Improve the Performance and Security in Cloud Using File Division Technique

Dr. N. V. Ramana Reddy[1], Srirama mariyamma[2]

[1]Associate professor , Department of CSE, Avanthi Institute of Engineering and Technology, Gunthapally (V), Abdullapurmet(M), RR District – 501512, Telangana, India

[2]Research Scholar, Department of CSE, Avanthi Institute of Engineering and Technology, Gunthapally (V), Abdullapurmet(M), RR District – 501512, Telangana, India

Abstract

Outsourcing data to external administrative control, as is done in cloud computing, raises security concerns. Data breaches can occur through attacks from other users or nodes in the cloud. Therefore, protecting data in the cloud requires advanced security measures. This project proposes fragmentation and replication in the cloud for optimal performance and security, addressing security and performance issues simultaneously. This method splits the file into fragments and copies the fragmented data between cloud nodes. Each node stores only a single fragment of a given data file, so even a successful attack does not leak meaningful information to the attacker.

Index : fragmentation, simultaneously, splits, data file.

Introduction

First of all Cloud computing is the use of computer resources (hardware and software) delivered as a service over a network (usually the Internet). The name derives from the common use of cloud-shaped symbols as abstractions for the complex infrastructure involved in system diagrams. In cloud computing, remote services trust your data, software, and computations. Cloud computing is hardware and software resources made available as a managed service by third parties over the Internet. These services typically allow access to sophisticated software applications and high-end networks of server computers.

1.1 Problem definition

In our project, we propose a "Data Security on the Cloud by Fragmentation" system that solves security and performance challenges.

Purpose The goal of this project is to fragment data in the cloud for security reasons and solve security and performance issues collectively. You need to be able to achieve economies of scale by reducing spending on technology infrastructure, increasing volume and productivity. Accessibility is improved by making information more accessible with minimal upfront costs. It is also said to be able to monitor data more effectively and provide flexibility and security for data uploaded to the cloud.

MotivationData centers used by cloud providers may also be subject to compliance requirements. Using a cloud service provider (CSP) can raise additional data sovereignty security concerns, as a customer's or tenant's data may not reside in the same system, in the same data center, or even in the same provider's cloud. I have. The proposed system implements data partitioning in the cloud for optimal performance and security, and addresses security and performance issues collectively. This method splits the file into pieces and puts the fragmented data in the cloud. Stores only a single fragment of a given data file, ensuring that even a successful attack does not leak meaningful information to the attacker.

Major Contributions of ProposedHere is the solution:

- Data-centric solution with data protection for cloud service providers
- No access.
- Rules-based authorization approach where rules are managed
- Data Owner.Very meaningful for authorization rules that enforce RBAC schemes with role

hierarchies and resource hierarchies (hierarchical RBAC or hRBAC).

• Access control computation was delegated to the CSP, but could not be granted.

• Unauthorized access.

• DataDepartment. The proposed solution could use data encryption to make the data inaccessible to the CSP or to bypass authorization mechanisms to release the data. Therefore, his aforementioned ABE-based solution proposed to solve access control in cloud computing is based on the attribute-based access control (ABAC) model. In addition, the proposed solution supports ontological representations of authorization models and provides additional reasoning mechanisms to deal with issues such as detecting conflicts between different authorization rules. The proposed solution is not tied to any PRE schema or implementation. To provide a comprehensive and workable solution, the rest of this document is based on the IBPRE approach and notation. Data-centric authentication solutions have been proposed to secure data in the cloud. It enables authorization management using a rules-based approach and provides advanced role-based expressiveness such as roles and roles. object hierarchy.

1.2 Scope

The scope of the project is to provide users with a platform for uploading files to the cloud and improve the performance and security of user-uploaded files. Scope of this methodology. Split the file into fragments and put the fragmented data on the cloud node. Each node stores only a single fragment of a given data file, so even a successful attack does not leak meaningful information to the attacker.

Literature survey

2.1 Survey Of Major Area Relevantto Project

Cloud computing technology requires users to entrust their valuable data to cloud providers. Concerns about the security and privacy of outsourced data are growing. To achieve scalable, flexible, and fine-grained access control to offsite data in cloud computing, this paper extends ciphertext policy attribute set-based encryption (ASBE) to enable hierarchical attribute set-based encryption. encryption (HASBE). It has a hierarchical user structure. The proposed scheme not only achieves scalability due to its hierarchical structure, but also provides flexibility and fine-grained access control in supporting complex attributes of ASBEs. Additionally, HASBE handles user revocation more efficiently than existing schemes by using multi-value assignments for access expiration. We formally certify the security of HASBE based on the security of CP-ABE scheme (Ciphertext-Policy Attribute-Based Encryption). Cloud computing has become the most in-demand technology in the IT industry over the past decade. Cloud computing started as a data offload technology and has since evolved into a new computing platform for all IT related activities. The advent of cloud computing to deliver business-critical applications has increased the value of the cloud. On the contrary, cloud security encounters a myriad of threats and vulnerabilities on various fronts. Security features such as access control, digital signatures, encryption and decryption are applied to the cloud environment to protect cloud data. This article provides a detailed examination of all variants of attribute-based encryption (ABE) access control technology available in cloud environments. Observations will be made regarding the use of ABEs and how access is granted. Various ABE techniques are compared,analyzed, and recommendations are provided for use in various cloud application deployments. As more and more sensitive data is shared and stored by third-party websites on the Internet, there is a need to encrypt the data stored on those websites. A drawback of data encryption is that it can only be selectively shared at a coarse-grained level (that is, by sharing the private key with other parties). We are developing a new cryptosystem for granular sharing of encrypted data. Wecall this Key-Policy Attribute-Based Encryption (KP-ABE). In our cryptosystem, ciphertexts are tagged with a set of attributes, and private keys are associated with access structures that control which ciphertexts a user can decrypt. We demonstrate the applicability of the construction to audit log information sharing and broadcast encryption. Our design supports private key delegation, including hierarchical identity-based encryption (HIBE).

2.2 Techniques And Algorithms Relavent W.R.Tproject

In the cloud, if an entire file is stored on one node, a successful attack on that node will compromise the security of the entire file. These systems can reduce replication time by replicating files across multiple nodes. However, this increases the attack surface and greatly increases the risk of data security. Therefore, on these systems, increased performance translates into decreased security. Our method does not store the entire file in a single node. A file is split into several fragments and stored in several nodes. Therefore, in the event of a security breach, no meaningful information is passed to the attacker. Cloud Manager performs the following functions: Receiving files. File encryption Divide-and-conquer algorithms for file fragmentation and file decryptioncombinefilesDownload the file to the user. divide-and-conquer algorithm The divide-and-conquer method divides the problem into subproblems similar to the original problem, solves the subproblems recursively, and finally combines the solutions of the subproblems to solve the original problem. Since divide-and-conquer solves subproblems recursively, each subproblem must be smaller than the original, and there must be a base case for the subproblem. We need to consider a divide-and-conquer algorithm consisting of three parts: Divide the problem into multiple sub-problems, which are smaller instances of the same problem. It tackles subproblems by recursively solving them. If they are small enough, solve the subproblem as the base case. Combine the solutions of sub-problems to solve the original problem.

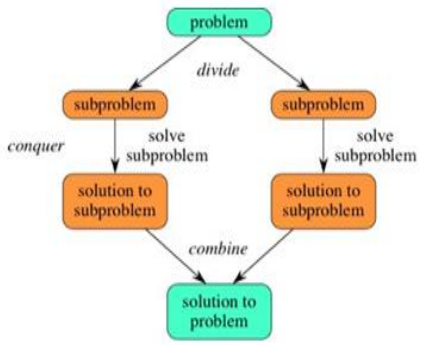


Fig 2.2 Overview of divide and conquer algorithm

2.3 Overview of AES algorithm

In the cloud, if an entire file is stored on one node, a successful attack on that node will compromise the security of the entire file. These systems can reduce replication time by replicating files across multiple nodes. However, this increases the attack surface and greatly increases the risk of data security. Therefore, on these systems, increased performance translates into decreased security. Our method does not store the entire file in her one node. A file is split into several fragments and stored in several nodes. Therefore, in the event of a security breach, no meaningful information is passed to the attacker. Cloud Manager performs the following functions: Receiving files. File encryption Divide-and-conquer algorithms for file fragmentation and file decryptioncombinefilesDownload the file to the user. divide-and-conquer algorithm Merge sort is a divide-and-conquer algorithm. Splits the input array in half, calls itself on its two halves, then merges the two sorted halves. The filespilttre() function is used to merge the two halves. Merge Sort is great for sorting linked lists, so that's what I use in my project. AES algorithm AES is iterative, not Feistel encryption. It is based on a "permutation permutation network". It consists of a series of linked operations, some of which replace an input with a particular output (permutations), and some of which involve bit rearrangements (permutations). Interestingly, AES performs all calculations on bytes, not bits. Therefore, AES treats 128 bits of plaintext block as 16 bytes. These 16 bytes are arranged in columns and rows for processing as a matrix. In contrast to DES, the number of rounds in AES is variable and dependent on key length. AES uses 10 rounds for 128-bit keys, 12 rounds for 192-bit keys, and 1 rounds for 256-bit keys. Each of these rounds uses a different 128-bit round key calculated from the original AES key.

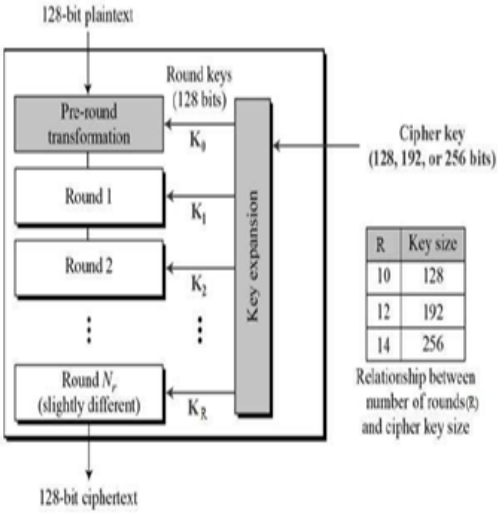


Fig 2.3 Overview of AES algorithm

2.4 Applications

educational application Cloud computing is very popular in the educational setting. Provide students with various online distance learning platforms and student information portals. The benefits of using the cloud in education are that it provides a powerful virtual classroom environment, easy access, secure data storage, scalability, extended reach to students, and minimal application hardware requirements. Ensuring the safety of stored data is therefore a major challenge and this can be achieved through the projects we propose.

Entertainment Applications The entertainment industry uses a multi-cloud strategy to engage with its audience. Cloud computing offers a variety of entertainment applications such as online games and video conferencing.

Online Gaming: Cloud Gaming is he one of the most important entertainment media today. It offers a variety of online games that run remotely from the cloud. Video conferencing apps: Video conferencing apps offer a simple, instantly connected experience. It allows you to communicate with business partners, friends and relatives via cloud-based video conferencing. The benefits of video conferencing are reduced costs, increased efficiency, and elimination of interoperability. iii. Management application Cloud computing offers a variety of cloud management tools that allow administrators to manage all kinds of cloud activities, such as: B. Resource provisioning, data integration, and disaster recovery. These management tools also provide administrative control over platforms, applications, and infrastructure.

3. System architecture

The owner/us er first uploads the fragment to the cloud. Files are divided into fragments based on some user criteria, and each cloud node (computer, storage, physical node, we use the term "node") so that each fragment does not contain meaningful information.) is saved. and virtual machines) contain their own fragments to enhance the security of your data. Node eligibility is determined by the selection of the most central node to provide better access times. It uses the concept of centrality to reduce access time.

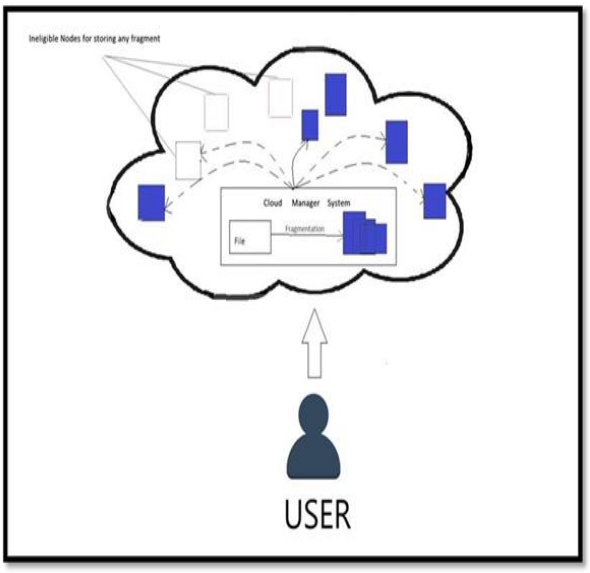


Fig 3.1: Overview of System Architecture

4. Implementation

4.1 Owner Module

The owner is the person who uploads the file to the cloud. Therefore, the data contained in the file is also stored in the cloud. Once data is moved to the cloud, self-protection packages are generated by data owners. This package contains encrypted data objects, authorization rules, and corresponding re-encryption keys. Data objects are encrypted before being uploaded to the cloud to prevent access by cloud service providers. This is done by the data owner using the encrypt() function. The data should be encrypted using the id ido1 of the uploaded object o1. Instead of direct

encryption, a digital envelope approach can be used to protect data objects. Authorization rules are defined by data owners and mapped directly to authorization models. This is done by including the appropriate elements in the binary relation.

4.2 Proxy Module

The following set of functions is provided by IBPRE. In this project, we will use encryption to protect data as it moves to the cloud. The authentication model is protected using advanced encryption techniques to prevent CSPs from disclosing data without the consent of the data owner. Specifically, this solution is based on Proxy Re-Encryption (PRE). The PRE scheme is an encryption scheme that allows an entity called a proxy to transcode data from one key to another without decrypting it. The proxy re-encryption scheme used for the proposal in this document requires the following features:

• Unidirectional. The one-way scheme allows us to generate the encryption key rkα−›β without allowing re-encryption from β to α.

• Not interactive. A non-interactive scheme allows users to create re-encryption keys without involving other entities.

• Can be used multiple times. Multiple-use schemes allow a proxy to perform multiple cryptographic operations on a single ciphertext.

That is, from u α u β , u β to c γ , etc. The master public key is publicly known and can be used directly by a user to generate another user's public key based on his girlfriend's identity. The master private key must remain secret and can be obtained from any trusted entity holding the master private key. This entity is called a Private Key Generator (PKG).Represents a proposed cryptographic primitive.

setup (p,k) —› (p, msk)

(1) keygen (p, msk, id) —› sk α

(2) encrypt (p, id_,m) —› c α

(3) rkgen (p, sk α , id α , idβ) —› rk α—›β

(4) Re-encryption (p, rk α—›β , c α) —›cβ

(5) Decryption (p, skα , c α) —› m (6)

For details on the cryptographic operations performed by these functions, see . A brief description of each function follows. Initialize the encryption scheme. User module A data-centric security approach should encrypt data to prevent unwanted access. An access control mechanism must then control who can decrypt the data and access its contents.

4.3 Encryption and Decryption

An architecture for deployment within CSP is also proposed. This architecture considers various elements that need to be introduced to provide an overview.How to access protected data using this approach. Once data is moved to the cloud, self-protection packages are generated by data owners. This package contains encrypted data objects, authorization rules, and corresponding re-encryption keys. Data objects are encrypted before being uploaded to the cloud so that CSP cannot access them. This is done by the data owner using her encrypt() function. The data should be encrypted using the id ido1 of the uploaded object o1. Instead of direct encryption, a digital envelope approach can be used to protect data objects. Authorization rules are defined by data owners and mapped directly to authorization models. This is done by including the appropriate elements in the binary relation.

To keep data safe in the cloud, the following conditions must be met: • CSP should not be able to access MSK.

• The CSP should not have access to the Authorization Factor's private key. If you use PKG, you need to make sure it doesn't conflict with CSP.

Data fragmentation The data file fragmentation threshold is generated by the file owner. File owners can specify the fragmentation threshold as a percentage or the number and size of various pieces. For example, a fragmentation ratio threshold might indicate that each fragment is 5 percent of the total file size. Alternatively, the owner can create another file containing the fragment number and size information (eg fragment1).

The size of fragment 2 is 5,000 bytes and the size of fragment 2 is 8,7 9 bytes. Claims that the file owner is the best candidate for generating a fragmentation threshold. The best way for the owner to split the file so that each fragment doesn't contain a lot of information is because the owner knows all the facts related to the data. A default fragmentation threshold percentage can be part of a service level agreement (SLA) if the user does not specify a fragmentation threshold

when uploading a data file. Our primary focus is storage system security. When a file is split into fragments, the proposed system chooses a cloud node to place the fragments. .3 Integration and Deployment • We have integrated all the modules into one system, but the main problem in integrating the modules is the encryption implementation. • The database integration and UI part was a challenge. I solved the problem by understanding the project flow

5 EV ALUATION PERFORMANCE

5.1 EVALUATIONPROCEDURE

We compared the performance of the algorithms. The behavior of algorithms is studied by:

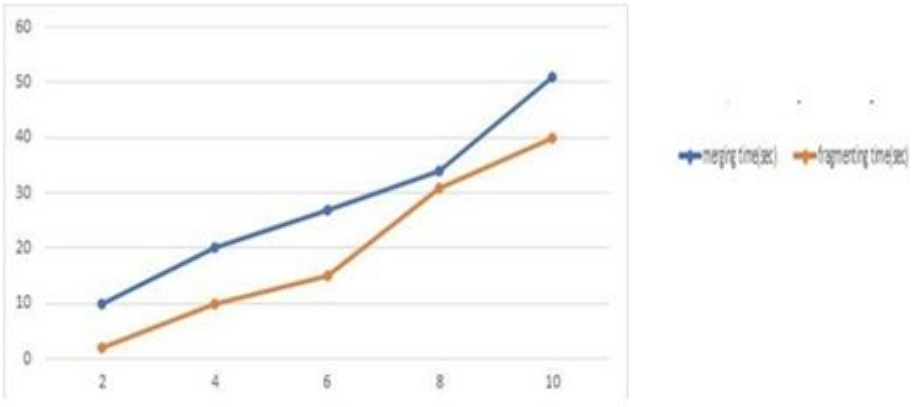
• increase the number of system fragments,

• increase the number of fragments, while the number of nodes remains unchanged,

• file resizing

• read/write mode variation.

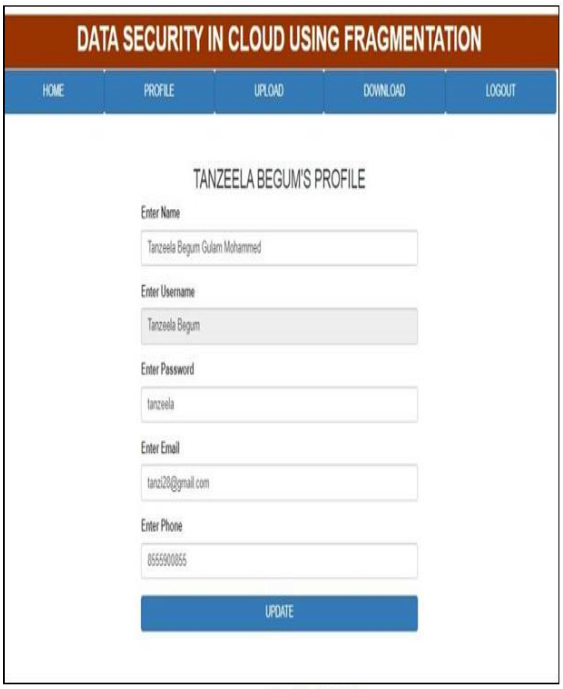
The mentioned parameters are important because they influence the size of the problem and the performance of the algorithms.



Impact of increasing number of file fragments

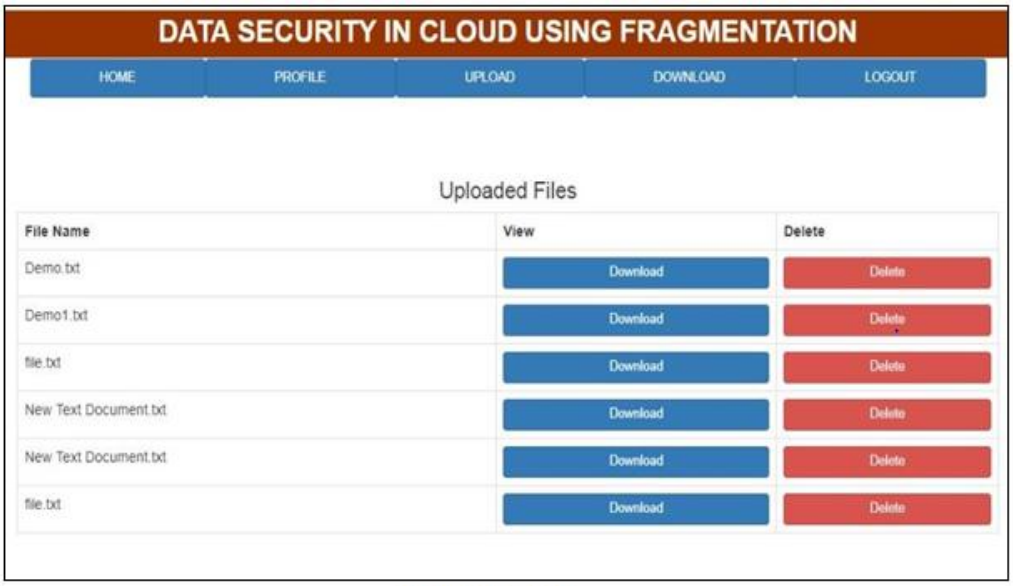
Registration form

User Registration Successful



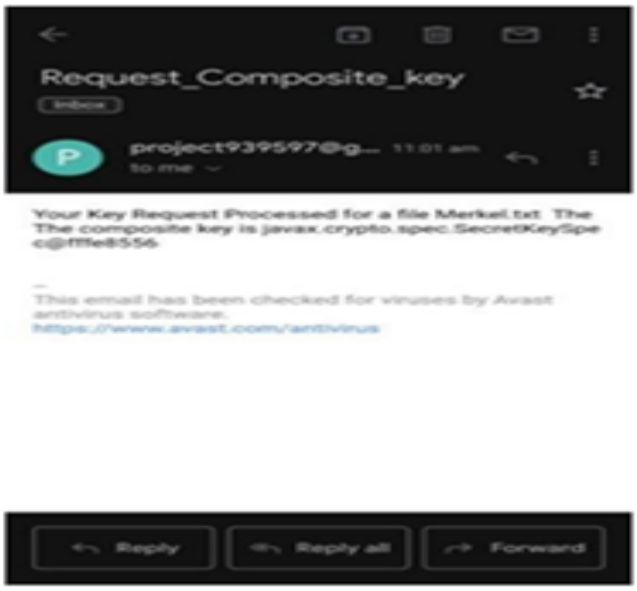
Profile Update

Upload File



Uploaded Files

Uploaded files



Request key

Composite key

Download file

6. Conclusion And Futureenhancements

In this project, we propose fragmentation of data in the cloud for security. The proposed system has been successfully implemented to improve the security of cloud data storage. In this method, we divide the file into parts, ensure encryption and store the fragmented data in the cloud. It stores only one part of a particular data file, which ensures that even in the case of a successful attack, the relevant information is not exposed to the attacker. We compared the effect of increasing the number of fragments on loading and unloading times. As a future work, we can implement mobile OTP verification and more encryption algorithms to improve security

7. References

[1] K. Bilal, S. U. Ĥano, L. Zhang, H. Lio, K. Hayat, S.A. Madani, N. Min-Allah, L. Wang, D. Chen, M. Iqbal, C. Z. Xukaj A. Y.Zomaya, "Quantitative Comparisons of Modern Data Center Architectures", Concurrency and Computing: Practice and Experience, Vol. 25, No. 12, 2013, pp. 1771-1783.

[2] K. Bilal, M. Manzano, S. U. Khan, E. Calle, K. Li ja A. Zoomaya, "On Structural Resilience Characterization of Service Center Networks", IEEE Transactions on Cloud Computing, Vol.1, numero 1, 2013, lk 6 -77.

[3] D. Boru, D. Kliazovič, F. Granelli, P. Bouvry and A.Y. Zomaya, "Energy-Efficient Data Replication in Cloud Data Centers", IEEE Globecom Workshops, 2013, pp. 446-451.

[4] Y. Deswarte, L. Blain and J-C. Fabre, "Intrusion Tolerance in Distributed Computing Systems", in Proceedings of IEEE Computer Society Symposium on Research in Security and Privacy, Oakland CA, p. 110-121, 1991.

[5] B. Grobauer, T. Walloschek and E. Stocker, "Komprenantenubkomputado vulnerabilities", IEEE Security and Privateco, Vol.9, no 2, 2011, pp. 50-57.

[6] W. K. Hale, "Frequency assignment: Theory and Applications", "Proceedings of the IEEE, Voi. 68, nro 12, 1980, s. 1 97-151 .

[7] K. Hashizume, D. G. Rosado, E. Fernndez-Medina ja E. B. Fernandez, "Analizo de sekurecproblemojpornubakomputado", Journal of Internet Services and Applications, Voi. , nr 1, 2013, s. 1-13.

[8] K.Bilal,M.Manzano,S.U.Khan,E.Calle,K.Li and A.Zomaya "OnDROPSmethodology"", IEEE Horizontal Pipe Propeller, 2010 .