Web Services for Mobile Devices from One Server

Mirsat Yesiltepe

Department of Mathematical Engineering, Yildiz Technical University, Davutpasa Road, 34220 Esenler- Istanbul

mirsaty@yildiz.edu.tr

***Abstract*— Mobile devices used for communication between people only for speaking people in the past but today one of the most personal kind devices in the cloud environment [1]. The environment is expanding day by day via accessibility connecting internet with devices. The device can be any kind of device if it has the ability working with web. The transaction should be with sending and receiving messages [2]. Mobile devices can connect the cloud also [3]. This article aims answering the question "Is cloud environment server its system equally for any kind of mobile devices which has different operation system?", throttling mechanism for mobile world which has the devices with multiple cores and push services for mobile devices. In this paper, will be discussed two mobile operations, which are Windows Phone and Android because of comparison of open source and secret sauce. Push services, the use of different protocols in the same environment with the resulting differences and message level security issues are another issue discussed in the paper.**

***Keywords*— mobile devices, message level security, throttling mechanism, protocol**

I. INTRODUCTION

Cloud computing has many meaning for everyone who work with the cloud. There are mainly three different definitions. One of definition is that it has been considered as one of the potential solutions to our increasing demand for accessing, processing, storing, and using provisioned resources over the internet [4]. Another definition is it is a new method to add capabilities to a computer without licensing new software, investing in new hardware or infrastructure or training new personnel [5]. Last definition, it is a service provider major challenging problem is designing efficient mechanism for managing the limited resources shared by different applications [6]. Although definitions reveal the concept of a common definition despite the emphasis on different aspects of the same concept. Why is the presence of a personalized description of the cloud concept? Because the concept of cloud for someone working on the same project with the concept of cloud for one may be different.

So how should cloud the definition of this paper? It will be utilized for airborne cloud concept in real terms. Cloud, is an environment where people take advantage to do the work of the relevant environment and how to do what is unknown and generally non-interest environment. For example, a user database is a cloud environment for users to know the status of the database where it is located by the bank transaction information stored when the money through a bank transfer. As trust in any environment is the most important security challenges in this environment [7]. There are a variety of mechanisms and concepts of this problem. The environment for the person in the cloud server and client are examples of known security problems in the count.

Another reason that the very definition of cloud concepts can expand and contract the cloud environment for people. In the past, the company data is stored by the company today began to be stored in places outside environment that provides this service. This will be the reason, the concept of cloud extended to these companies.

Cloud environment is concept of implementation, management, and security [8]. When using the environment creator and user of the environment had to think the manners.

One of the concepts discussed in the cloud environment is whether is there any difference a classic in defining the concept of client and server. In general, the server is responding party in client requests. However, today, responding to some of the client and the server are the clients of other servers that may be the architect of the circular [9].

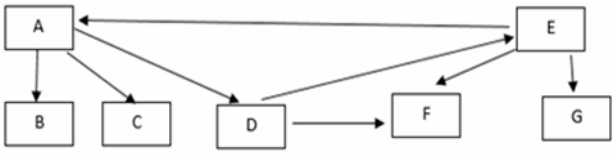


Figure 1. Client – server architecture example situation (→ request to response direction)

In Figure 1 it is shown in this case. In this environment devices, can be client or server for different conditions. For example, A is a server for B, C, D but a client for E. In this paper, said cloud is like unknown environment. In this figure A request information from D and E request the information form E which A request. A do not know E so for requesting information D, E is in cloud environment. Same state can be with A, D, F devices. A do not know F.

II. PUSH SERVICES

Push service is a safe and effective way to alert the user. Even the background applications, allows to inform users in real time. Push service is widely used in weather updates, messaging services, email notification, including applications such as mobile coupon services are used in various fields. Push services are optional, but it has become necessary. It has three main pre-request.

1) Permission must be obtained for the use of the push service.
2) Push messages should be sent with the POST method.
3) Post requests should use debugging errors, state codes and messages [10].

A. Windows Phone Push Services

Sending a push notification process consists of three basic steps:

1) Channel request: WNS (Web notification service) will benefit from Winrt (Windows runtime) API (Application programming interface) to request a channel URI (Uniform resource identifier). URI channels to send notifications to your application will be a unique identifier that will used.
2) Windows Azure Mobile Services channel recording: If the channel can then store the channels and service are until you decide it is time to send a notice to the channel given that any application-specific data (e.g. user profiles) can be associated with it.
3) Authentication for WNS and push notification

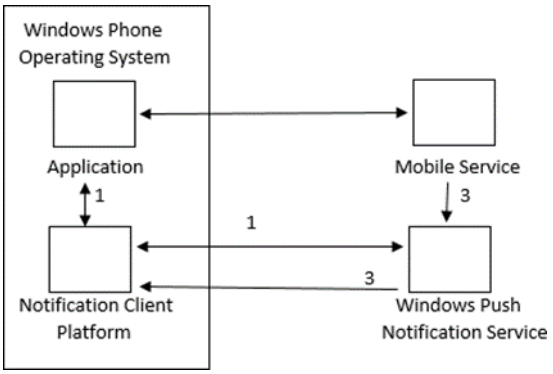


Figure 2. Windows Phone based push service mechanism [11]

B. Android Push Services

Overall enrollment service to push the use of the service at the push of Android, the issuing of permits, selecting the icon thrust, adding parsing API key, enable push notifications, send the test can be summarized as push notifications.

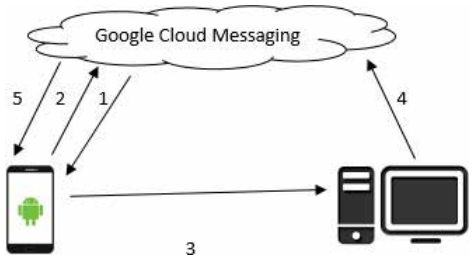


Figure 3. Android based push service mechanism

1) When the ADF (Application Development Framework) mobile application is started, sends a registration request with the push notification service. Sender sends recording qualifier GCM (Google Cloud Messaging) server for recording Android devices.
2) After successful registration, the GCM servers, ADF sends a recording qualifier mobile applications and devices.
3) After receiving the ADF tray mobile find records qualifier application life cycle events OnOpen method is called in listeners.
4) Provider application server, the database stores registration qualifier for later use.
5) GCM server log messages are distributed tray using qualifiers [12].

Push service are used by mobile devices is a member of cloud computing [13]. Push service in the two-mobile operating system in the event loop to connect to make love clouds are similar. However, the connection to the cloud operating system is willing to use their programs. Operating systems at tasks in the push service that they would like to use their trust.

III. THROTTLING MECHANISM

While it is not a direct instance management technique, throttling enables you to restrain client connections and the load they place on your service [14]. With the mechanism server, can decide how one or more clients can do the whole work. The following test is researched that. In this session, did not given detail information. For more information, you can read the paper [15].

In the test scenario, the environment has a single server and one or more client for different size of work. The work consists of sending a sentence and waiting for one second. Test results are showed in table 1 to 4 which are in tested in different conditions. Numbers are working second time.

TABLE I. ONE CLIENT 20.000 SIZE WORK

Concurrency Mode	Instance Context Mode		
	Single	***Per Session***	***Per Call***
Single	477	20	20
Multiple	21	20	21
Reentrant	504	21	21

TABLE II. ONE CLIENT 500 SIZE WORK

Concurrency Mode	Instance Context Mode		
	Single	*Per Session*	*Per Call*
Single	477	20	20
Multiple	21	20	21
Reentrant	504	21	21

TABLE III. 100 CLIENTS 20.000 SIZE WORK

Concurrency Mode	Instance Context Mode		
	Single	*Per Session*	*Per Call*
Single	-	342	147
Multiple	148	151	340
Reentrant	-	363	339

TABLE IV. 100 CLIENTS 500 SIZE WORK

Concurrency Mode	Instance Context Mode		
	Single	*Per Session*	*Per Call*
Single	508	20	17
Multiple	21	20	20
Reentrant	504	20	20

It can be summarized the results obtained from the above table as follows. For all conditions, worst are single – single and reentrant – single. The best is single – per call condition.

TABLE V. SUMMARY TABLE

Client count / Work size	Small	Big
One	Multiple - *	Multiple - *
Multiple	Multiple - *	Multiple – Single, Multiple PerSession

In big work if client number increased for same work, single – per session, reentrant – per session and reentrant – per call conditions are getting worse. For one client if the work size increased multiple – single, reentrant – single conditions getting worse.

From all above test environment, it can be summarized if working size is decreased for one client and managing the work is getting harder. For smaller work, no condition was get worse. If work size is increasing two condition only get worse others are not so do not use the condition for every work.

There is no difference between tested two mobile operation systems.

IV. MESSAGE LEVEL SECURITY

Message level security can provide end-to-end secure guarantee in a heterogeneous environment [16]. This security in environments different from others is even though more than one client and one server to a server that is working to ensure the security and the successful communication between clients. Message security is the only option to provide security when you have intermediate routers to route request / response [17].

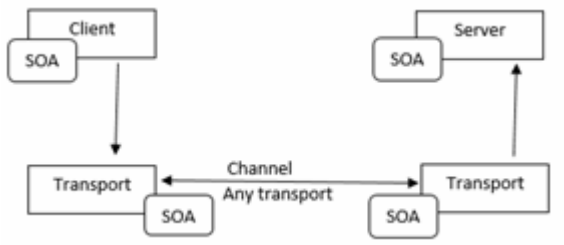


Figure 4. Message level security mechanism of testing environment [18]

This level of security is not altogether the communication with the others one by one through the server to the client is interested in the security factor and that the communication is. The security cannot be achieved when the intermediate channel communication channel is completed and try to establish a new channel through like communication.

TABLE VI. MESSAGE LEVEL SECURITY ENCRYPTION ALGORITHMS

1-Basic128	9-Basic256
2-Basic128Rsa15	10-Basic256Rsa15
3-Basic128Sha256	11-Basic256Sha256
4-Basic128Sha256Rsa15	12-Basic256Sha256Rsa15
5-Basic192	13-TripleDes
6-Basic192Rsa15	14-TripleDesRsa15
7-Basic192Sha256	15-TripleDesSha256
8-Basic192Sha256Rsa15	16-TripleDesSha256Rsa15

Messages are transferred with either of unsecure, encrypted, signed format. If encrypted format is chosen, it must be used with either of encryption algorithms in Table 6. The encryption algorithm names and security levels of encryption algorithms are ascending order.

V. MESSAGE LEVEL SECURITY ENVIRONMENT

This section of the test environment server will examine the steps followed in establishing client for Windows Phone and Android client. This is a cloud service objective in the test environment will be created for different operating systems and the way I connect to this cloud is monitoring the communication with the media. An exemplary code snippet tested is a condition code obtained particles. Change of test cases has led to a change of the code snippets. For web services is used WCF (Windows Communication Foundation).

A. *Creating the Server in The Test Environment*

The following steps were applied to the test phase of the build server:

1) First created interface will determine only the names of roads and services offered by the server, this interface was created based on the class of service.
2) The communication between the client and the server will assume the duties App.Config bridge configuration file is created. The configuration file,

per the type of encryption algorithm used test cases, the server is updated with the new environment in case of a change of binding used is provided.

3) Service file is created. The services to be offered by service parameter (service) is specified position where.

4) For ISS, WAS publishing method is chosen because of the acceptance of the publication of the TCP protocol server. WAS is the process activation mechanism introduced in version 7.0. The reason for not selecting the Windows service delivery methods is to provide the Android Clients to connect to the environment

Thus, ensured the participation of the TCP protocol in place can be used in the Android client environment, other cases are provided to join the Android client only when using the HTTP protocol environment.

B. *Creating Windows Phone Client in The Test Environment*

Application service on the server as a service reference (to serve the service by specifying the URL (Uniform resource locator) address) is added. This URL is accessed through the corresponding WSDL (Web service description language) file.

Application service on the server as a service reference (to serve the service by specifying the URL (Uniform resource locator) address) is added. This URL is accessed through the corresponding WSDL (Web service description language) file.

C. *Creating Android Client in The Test Environment*

Web service reference is connected to the server via the URL service. In fact, the service created by connecting the server with the URL provided via the WSDL file. To be laid on information generated by the WSDL file server where only a portion of the file is shown. WSDL file of the client and can be considered as a contract where the protocol for communication between servers.

Services related to informing the service application page is used in practice.

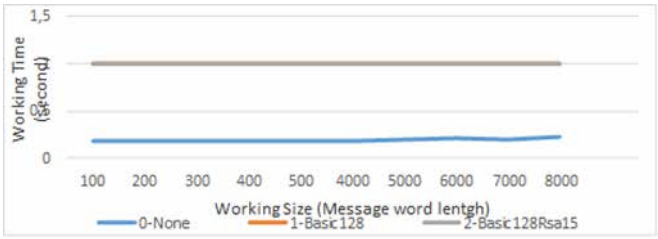
In two kind client environment term, multiple means ten clients.

VI.I. MESSAGE LEVEL SECURITY ENVIRONMENT RESULTS

The tables below are shown in the table occurs in different situations. In below tables in this section the other algorithms near BasicRsa15 values.

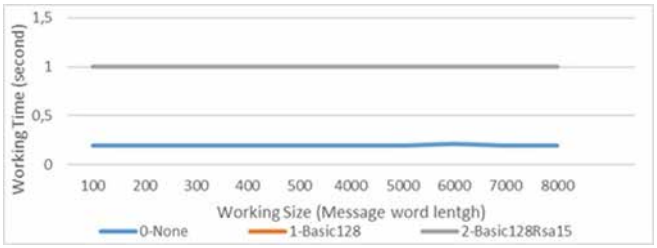
A. *One Client and One Server Environment*

TABLE VII. ONE CLIENT AND ONE SERVER ENVIRONMENT WITH TCP PROTOCOL



In table 7 with insecure conditions in the communication, time is around 0.2 milliseconds. Time, in environments created with encryption algorithms that are about one millisecond. Increasing the number of characters in the SOAP protocol communication message communication time per character in the environment has led to even more efficient. However, the yield efficiency of communication messages in this environment without increasing the number of characters is gradually decreased.

TABLE VIII. ONE CLIENT AND ONE SERVER ENVIRONMENT WITH HTTP PROTOCOL



In table 8 with insecure conditions in the communication, time is around 0.2 milliseconds. Time, in environments created with encryption algorithms that are about 1 millisecond. Increasing the number of characters in the SOAP protocol communication message communication time per character in the environment has led to even more efficient. However, the yield efficiency of communication messages in this environment without increasing the number of characters is gradually decreased.

B. *Multiple Clients and One Server Environment*

TABLE VIII. MULTIPLE CLIENTS AND ONE SERVER ENVIRONMENT WITH TCP

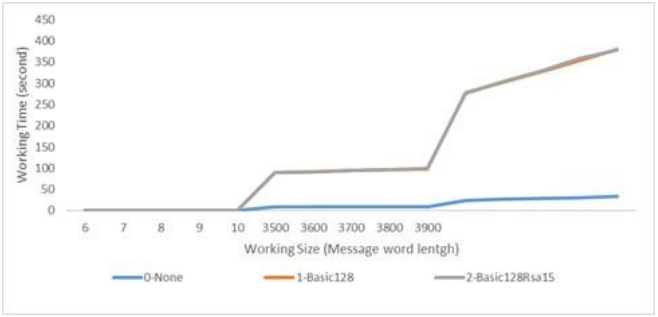
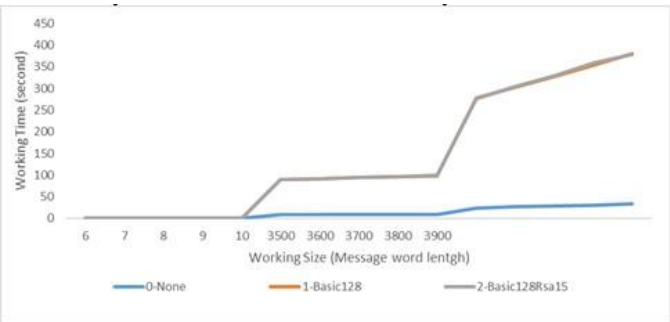


Table 9 when examined in an environment that is unsafe and encrypted communications the contact time has been

observed that when increasing the number of communications increases linearly. The values obtained in an encrypted environment with insecure conditions has steadily increased. The number of communications in environments where there are unsafe and encrypted communications decreased the efficiency of the communication time is increased is fixed at a certain level. In Asymmetric encryption algorithms with established test environment due to the efficiency of the communication time communication time when you could get very little asymmetric negative values.

TABLE X. MULTIPLE CLIENTS AND ONE SERVER ENVIRONMENT WITH HTTP PROTOCOL



In table 10 when examined in an environment that is unsafe and encrypted communications the contact time has been observed that when increasing the number of communications increases linearly. The values obtained in an encrypted environment with insecure conditions has steadily increased. The number of communications in environments where there are unsafe and encrypted communications decreased the efficiency of the communication time is increased is fixed at a certain level. In Asymmetric encryption algorithms with established test environment due to the efficiency of the communication time communication time when you could get very little asymmetric negative values.

VII. MESSAGE LEVEL SECURITY RELATIONSHIP TABLES IN TEST ENVIRONMENT

Per this section, the relevant tests with variables tested encryption algorithms that are grouped close together performance.

TABLE XI. ONE CLIENT AND ONE SERVER ENVIRONMENT WITH TC PROTOCOL

Basic128, Basic128Rsa15, Basic192
Basic128Sha256, Basic128Sha256Rsa15, Basic192Sha256, Basic192Sha256Rsa15, Basic256, Basic256Rsa15,TripleDes, TripleDesRsa15
Basic256Sha256, Basic256Sha256Rsa15
TripleDesSha256, TripleDesSha256Rsa15

When a single-client TCP protocol - linking environment to examine the general summary of data values obtained while communication across unsecured environments ranging messaging sizes specified time communication based algorithms passwords changing, while the fastest has not changed. Total parameters of communication messages, while unsecured medium 2 has received the message does not change but 10 value per the tested algorithms. Contact message size value when in insecure conditions at least this time of the communication received by the encryption type of message size value is seen differently

TABLE XII. ONE CLIENT AND ONE SERVER ENVIRONMENT WITH HTTP PROTOCOL

Basic128, Basic128Rsa15, Basic192, Basic192Rsa15, Basic192Sha256, Basic192Sha256Rsa15, Basic256, Basic256Rsa15, TripleDes, TripleDesRsa15
Basic128Sha256, Basic128Sha256Rsa15
Basic256Sha256, Basic256Sha256Rsa15, TripleDesSha256, TripleDesSha256Rsa15

Single-client environment must connect when HTTP protocol general summary of data values obtained in insecure conditions in the period studied communications across messaging varying sizes are unchanged, while the fastest communication time by changing encryption algorithms specified sensitivity. Contact message units, while in the insecure conditions in parameters two has received the message does not change but 10 values tested by the algorithm. Contact message size value when in insecure conditions at least this time of the communication received by the encryption type of message size value is seen differently.

TABLE XIII. MULTIPLE CLIENTS AND ONE SERVER ENVIRONMENT WITH TCP PROTOCOL

Basic128Sha256
TripleDesRsa15, TripleDesSha256
The Others

Multiple-client environment must connect when TCP protocol general summary of data values obtained in insecure conditions in the period studied communications across messaging varying sizes are unchanged, while the fastest communication time by changing encryption algorithms specified sensitivity. While communication message number parameter in insecure conditions at two, took the message does not change but 10 values tested by the algorithm. Contact message size in the insecure conditions in which this time takes a minimum value when the size of the communication message per the type of encryption those different values were observed.

TABLE XIV. MULTIPLE CLIENTS AND ONE SERVER ENVIRONMENT WITH HTTP PROTOCOL

Basic128Rsa15
Basic128Sha256
TripleDesRsa15
TripleDesSha256Rsa15
The Others

Multi-client environment must connect when HTTP protocol general summary of data values obtained in insecure conditions in the period studied communications across messaging varying sizes are variable communication time by changing encryption algorithms, while the fastest. Total parameters of communication messages, while unsecured medium 2 has received the message does not change but 10 value per the tested algorithms.

VIII. CONCLUSION

The tested two mobile operation system for any communication with the cloud there is no problem. However, in some case, they do not trust each other and connecting a cloud Android give values that are more flexible.

In mobile and cloud environment HTTP protocol is used compared with TCP so today application is more software than hardware. The cause of making this known, actual work, although it is difficult to adapt to cloud environments TCP aim of using the protocol is the comparison of the factors that make less use Although a faster protocol. Therefore, the speed difference will be measured in the tests is to find the ratio of the preferred next improvable quality requirements in this context.

While the default encryption type of message-level security Basic128 in HTTP protocol, the type in TCP protocol is insecure. Both protocol support message-level security encryption types.

The cloud environment is still not fully suitable for use TCP protocol. TCP is faster communication protocol even though the idea of carrying the message may be lost parts of this protocol reduced the utilization rate in the cloud environment. The HTTP protocol to guarantee the situation and identify the most system has increased the use of this protocol is in the cloud environment.

The message-level security has linear complexity. Operating speed and size of the program in the face of the increasing number of communication messages as constant increases. However, this time, because the number of messages in the same communication pattern size is increased once the communication message communication message size has not changed too much. Because edit tags in the communication message size in SOA (Service Oriented Architecture) communication message (except for the labels used in all messaging message) size is not too much change too much this variable. Which can be used in mobile applications, such as GPS support but do not support corporate PCs plus providing an advantage for mobile applications is just an example of a condition that causes the application to be responsive to the mobile client. This causes the server therefore not give the same answer to any kind of client.

SOAP (Service Oriented Architecture Protocol) communication made in accordance with the rules on the tested service request has been observed to be inadequate for enterprise applications, the default value. As an example, the default communication connection time is one minute. These values are inadequate for enterprise applications. Contact time may be up to about an hour.

When the SOAP message level security asymmetric encryption algorithms running, time compared with symmetric encryption algorithms based on the number of message communication has been shown to be more erratic. Therefore, using a SOAP message-level security in symmetric encryption algorithm will be better.

In mobile communication is continuing today, every moment can be detached from the idea of the client's media were encouraged to use REST (Representational State Transfer) instead of using SOAP. Using REST, the advantage is a shorter form of communication messages (without adding more titles) is that whether the communication environment.

Today, sided asynchronous data transmission in mobile communications, the corporate side synchronous communication is another proof that the use of cloud environment is not fully formed.

The conclusion is to be reached when the results are collected. If an improvement to the desired application without the enterprise application where appropriate the use of the HTTP protocol, but the application enterprise if it is advantageous for the HTTP protocol to use in interacting with the cloud, but organizations in the communication between client and server is achieved thus may be advantageous TCP protocol. Because of communication within the tested application will not be a waste of corporate messages. As it observed by the absence of any data loss on testing.

REFERENCES

[1] Kleinman, Lisa, Tad Hirsch, and Matt Yurdana. "Exploring mobile devices as personal public displays." Proceedings of the 17th No. 5,634,127. 27 May 1997.

[2] Huerta-Canepa, Gonzalo, and Dongman Lee. "A virtual cloud computing provider for mobile devices." Proceedings of the 1st ACM Workshop on Mobile Cloud Computing & Services: Social Networks and Beyond. ACM, 2010.

[3] Gugnani, Gunjan, et al. "Implementing DNA Encryption Technique in Web Services to Embed Confidentiality in Cloud." Proceedings of the Second International Conference on Computer and Communication Technologies. Springer India, 2016.

[4] International Conference on Human-Computer Interaction with Mobile Devices and Services. ACM, 2015.

[5] Cloud, Donald M., et al. "Methods and apparatus for implementing a message driven processor in a client-server environment." U.S. Patent

[6] Rewatkar, Liladhar R., and U. L. Lanjewar. "Implementation of cloud computing on web application." International Journal of Computer Applications 2.8 (2010): 28-32.

[7] Bhaskar, R., and B. S. Shylaja. "Knowledge Based Reduction Technique For Virtual Machine Provisioning In Cloud Computing." International Journal of Computer Science and Information Security 14.7 (2016): 472.

[8] Pearson, Siani, and Azzedine Benameur. "Privacy, security and trust issues arising from cloud computing." Cloud Computing Technology and Science (CloudCom), 2010 IEEE Second International Conference on. IEEE, 2010.

[9] Rittinghouse, John W., and James F. Ransome. Cloud computing: implementation, management, and security. CRC press, 2016.

[10] Chernyi, Sergei. "The implementation of technology of multi-user client-server applications for systems of decision making support." Metallurgical and Mining Industry 3 (2015): 60-65.

[11] Bada Developer (2014), “Push Messaging Guide”, bada Master, http://developer.bada.com/article/push-messaging-guide.

[12] Deepak C Siddappa (2013), “ADF Mobile Push Notification With Google Cloud Messaging (GCM) Part 1”, Unwinding ADF.

[13] Nick Harris (2012), “What are Push Notifications?”, Azure Mobile Services.

[14] Chen, Wei, et al. "On Measuring Cloud-Based Push Services." International Journal of Web Services Research (IJWSR) 13.1 (2016): 53-68.

[15] Lowy, Juval. Programming WCF services. " O'Reilly Media, Inc.", 2007.

[16] Porter, George, and Randy H. Katz. "Effective web service load balancing through statistical monitoring." Communications of the ACM 49.3 (2006): 48-54.

[17] Tang, Wei-Dong, And Yong-Quan Zhou. "Message-Level Security Model For Web Services And Its Security Evaluation [J]." Computer Engineering And Design 10 (2006): 050.
[18] Medhi, Subhash, Abhijit Bora, and Tulshi Bezboruah. "Security Impact on e-ATM Windows Communication Foundation Services using Certificate based Authentication and Protection: An implementation of Message Level Security based on. NET Technique." International Journal of Information Retrieval Research (IJIRR) 6.3 (2016): 37-51.
[19] Medhi, Subhash, Abhijit Bora, and Tulshi Bezboruah. "Security Impact on e-ATM Windows Communication Foundation Services using Certificate based Authentication and Protection: An implementation of Message Level Security based on. NET Technique." International Journal of Information Retrieval Research (IJIRR) 6.3 (2016): 3 .

Mirsat Yesiltepe. He was born in 1989 in Istanbul and graduated from Yildiz Technical University, Department of Computer Engineering in 2015. The master thesis is "Evaluation of encryption methods used in service-oriented architecture". The departments that he under graduated are Computer Engineering and Finance. Currently, he is a research assistant at Yildiz Technical University, Department of Mathematical Engineering, and Systems Analysis subarea and continues his PhD education in the same department.