Journal of Engineering Technology and Applied Sciences, 2016 *e-ISSN:* 2548-0391
Received 13 January 2016, Accepted 26 March 2016 *Vol. 1, No. 1, 19-27*
Published online 27 April 2016 *doi: will be added*

Citation: Yeşiltepe, M . "Cloud Interaction and Safety Features of Mobile Devices". Journal of Engineering Technology and Applied Sciences 1 (1) 2016 : 19-27

CLOUD INTERACTION AND SAFETY FEATURES OF MOBILE DEVICES

Mirsat Yeşiltepe

Department of Mathematical Engineering, Yıldız Technical University, 34220, Istanbul, Turkey
mirsaty@yldiz.edu.tr

Abstract
In this paper, two current popular mobile operating system, still in relation to the concept of cloud began to supplant the internet almost Word today, the differences, the concept of cloud security mechanisms they use for themselves and are dealt with in this environment. One of comparing mobile operation system is representing open source and the other for close source one. The other issue discussed in this article is how the mobile environment interacts with the cloud than the cloud communication with the computers.

Keywords: Mobile device, cloud definition, pooling, security.

1. Introduction

Within a few years of looking at the progress of cloud services, devices are expected to communicate with data without the need of storage. The goal is to try to choose the best one per various criteria in the local issues by examining how different platforms support these cloud services, their differences from other services, security mechanisms, independent of the platforms used in the platforms and platforms used. Each platform will have advantages and disadvantages over its own. The goal here is to achieve the best by mentioning the advantages and disadvantages.

Mobile devices are the highest device when considering the rate at which the internet is used [1]. This can be said from the perspective that the REST (Representational state transfer) architecture used by mobile devices that have been maintained in a different way includes other web service architects.

In the future, the REST architecture will be equalized with SOA (Service-oriented Architecture) in every respect (not just in terms of syntax), perhaps the world of computers will leave the mobile world. But a problem will always exist. Yesterday was a security question. Security is a problem today. Tomorrow, security will be a problem. Because the developments in cloud

services allow the creation of new security mechanisms, they allow vulnerable users to find security vulnerabilities faster. This race will continue in the future

2. Cloud definition for mobile devices

Before describing the platform-like differences and similarities between mobile web services and mobile devices, cloud definition of platforms has been discoursed.

2.1 In terms of Windows Phone operating system

The concept of cloud is looking at Windows Phone as a floor covering such as OneDrive (Microsoft-side data storage) on the internet. What can be stored is anything that can be stored as a file, such as pictures, music, videos, etc. This information can be accessed from any device that has an Internet connection, such as a telephone or a television [2].

Figure 1. Cloud interaction with Windows Phone

2.2 In terms of Android operating system

An Android operating system. Unified operating system. Because there is no cloud definition specifically for Android, it uses the Google Cloud platform concept. This is the whole cloud-based service that Google Cloud Platform users can create as many sites as they can from simple sites to complex sites. With Google Cloud Platform, a back end for mobile devices can be created easily. New mobile experiences amateur replica kit. It can be developed rear ends with increasingly appropriate size. In doing so, problems such as load balancing and machine management are not considered [3].

Figure 2. Google Cloud Platform (only Android and IOS mobile support available) [3]

There are differences and similarities in the appearance of the two platforms. Their differences can be summarized as follows. The utility programs of the two are different. For example, they both store their data and do it with their own programs. They want to think of the cloud platform just like their environment. They do not want to talk about their compatibility with other platforms.

Similarities are like these. On both platforms, cloud thinks like a data storage space. The data may be stored publicly, with a certain cut open or hidden. Both are compatible with devices with internet connection. Users are only interested in what they want to do. There is no news in the cloud. They want clients and servers on the platform to be online. Both are a collection of cloud-based services they have in themselves.

The meaning of clouds that platforms load on the concept of cloud emerges the concept of definition cloud.

Cloud is the distributor of cloud services over the internet [4]. Cloud services are for individuals and businesses to use software and hardware managed by third parties in remote locations. Examples of cloud services include online file storage, social networking sites, web mail. Cloud services are used to access information on computers and resources when network connectivity is appropriate.

In this section, the concept of cloud will be discussed in terms of target users.

Private Cloud: The cloud infrastructure is only operated for a specific organization and is managed by an organization or a third party [5]. The user's mail information can only be accessed from where person wants it.

Community Cloud: The service is shared by various organizations and is only adapted to these groups. Infrastructure may be owned and operated by organizations or by a cloud service provider. Allows the user to see cloud information for everyone.

Hybrid Cloud: Combination of the methods in the welding pool. It is the type of cloud in which some information of the user is shared only by himself and some information is shared openly.

3. Communication with the cloud in terms of mobile devices

In this section, how the mobile device communicates with the server in different media will be processed. Often mobile devices are in an environment and communicate with each other and sometimes with cloud resources. Communication with the cloud is the topic of this episode.

To talk to the cloud, a communication protocol must first be supported by choosing a communication protocol and communicating with the cloud.

The communication protocol uses well-defined formats for message exchanges. There is something about a message, a full understanding of what is meant to be given and which can be removed. For this reason, it is essential to define a protocol syntax, semantics and communication synchronization in a protocol. It is typically independent of what will be applied with the behavior specified.

A protocol can be applied only to software or hardware, or both. The parties should agree on communication protocols [6]. To reach agreement, a protocol must be a technical standard.
A programming language uses the same states (semantics, syntax, and timing) for calculations. For this reason, there is a close resemblance between protocols and programming languages. Both are used for communication [7].

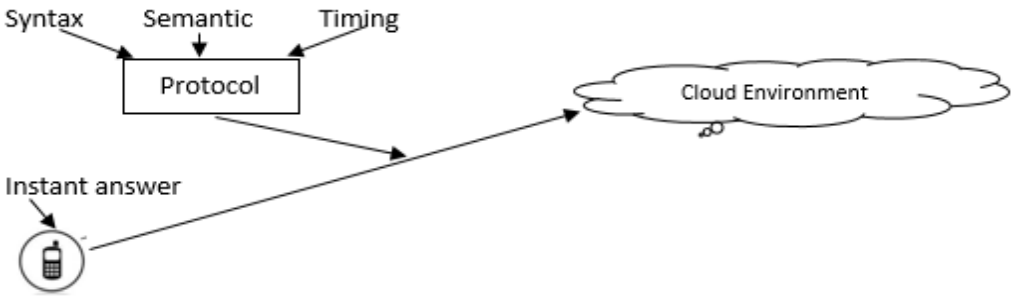

Figure 3. Protocol objectives diagram in mobile environment

There was a problem that the negotiations would be over when the old one had gone and the signal was cut off. It is an important question to determine the time-out when the cloud is set. One of the problems that arise from timeouts is when you will be prompted to authenticate again when connected to a website. Because of the time-out problem, the protocols are useful for remote, secure, reliable and rapid communication.

The reasonable period in which communication is accepted varies per the median. For example, if it is desired to communicate with space, it may be acceptable for the communication to take a few minutes, which is not acceptable for mobile devices.

The protocol identifies the syntax of how to communicate with other entities and determines the formats of messages that are to be sent to other entities. That is, when a message is sent, the protocol must have properties and syntax. Defines the semantics of messages by determining their semantics. When a client sends a message on this site, the opposite party can interpret the message correctly if it knows the semantics. Otherwise, this message is meaningless for the other party.

4. Mobile protocol layering

Layering is to partition the network system as large as possible. These parts are equipped by the service provided by the previous element alone [8].

Layering allows to make specific hardware for each layer: packet routing in the physical layer, packet routing in the IP layer Routers are made by the NATs in the transport layer, and packet routing in the transport layer. On this side, high TCP / IP performances can be obtained with simple hardware [9].

The most important disadvantage is that extra layers of information, such as being sure of the correctness of the work to be sent from one layer to the other layer, are the most important advantage in the layering of special hardware for each layer.

In the mobile world, the HTTP protocol is used for communication. There is TCP / IP layering in the protocol. There can be different protocols at the bottom. But the superstructure is generally the same. In other protocols, specific to mobile applications using HTTP, this layering may also be in the upper order. The purpose of these protocols is to be able to derive new protocols by adhering to HTTP. Essentially other protocols in the layer take HTTP as an example. So, HTTP is a reference protocol in mobile communication.

The main reason for using protocol layering in mobile communication is to change the format of the data in the communication between the client and the server and to ensure that there is no problem connecting to the cloud. So basically, the format will change but there will be no problem connecting to the cloud through HTTP which is the reference protocol.

HTTP design methodologies are two basic types in the mobile world. These are SOAP and REST and are discussed in detail in the following sections. But here is enough to know. Both methodologies can be used to communicate between the client and server in the cloud environment. SOAP is an older method. Despite the increasing use of both. In recent years, the rate of increase of REST in all these methodologies has been so high that SOAP has decreased in the overall percentage.

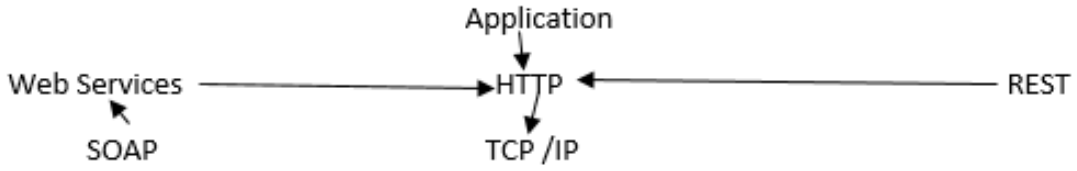

Figure 4. Mobile protocol layering

5. HTTP pooling for mobile devices

Mobile devices use the client-based HTTP protocol primarily for communication purposes. In the communications that are seen so far, clients, such as mobile devices, start by sending a request to the server to initiate the communication and the communication is executed to use the specific resources in the server per the request of the clients.

This section is the waiting period when you want to use the data associated with the resource in the server. So, in any case, how long will it be waiting for an answer to be sent when data update is requested? This time is important because it will shape the request to return new data per the incoming response from the server.

Let's assume that the server is communicating with many clients. The server wants to send a request to the second client. In case this message contains important information, it is requested that the message sent to the server be known by the client. Sometimes the clients want to update their data from the server per the future response. The communication between the client and the server may be initiated at the request of the client and updating the data may be encountered. The main problem with this section is that if the client sends a request in response to the request from the server and the server sends a request again per the future, and if the client does not reach the corresponding response, how long the client should wait and how long the server will respond when the server responds to the client Strategies have been examined to prevent wasted using.

The problem is that the client (the user) decides when to give a solution from the server. This method is widespread about the use and update of the user interfaces specified by the list images contained in the data. It is not always appropriate for the client to be informed of the server's future waiting period. The hardest part of this question is the time when the client will send a new request per the expectation of the server. This is a good way to get data from the server. The client selects the update job time itself. Another feature of this method is that the update can be used automatically when the update is done automatically or when the application is running and the data needs to be synchronized on the server. These are the purposes and benefits of using the event management model in the interaction of the client with the server. Failure of the client to leave the environment and the status can be renewed for a certain period. But at the end of a certain client, the client stops updating its status and disconnects from the environment. In summary, in this method, the client initiating communication with the server chooses not to communicate with the server for if it does not respond.

Another model that can be used to keep up-to-date clients with server information is the pooling model. The idea here is that when the server is ready, the client will start communicating. If the application is running or the service in the application is ready, the client will request server-supplied data at certain intervals. The client specifies time intervals. Requests to the server can be renewed at client-selected intervals. This is because the client does not know when the request is sent back and the data is not returned and why the situation is occurring. Here, the request is left to the server by sending all the requests to the server in the hope that the server will respond only in a blind manner. However, the resources on the server are wasted since requests are always sent to the server and the response is not returned and the resources are not available for the new client on the server. For this reason, the suitability of the client must be checked. If nothing happens, the answer should be returned. Resources include network resources, CPU (Central Processing Unit), and memory and should be considered.

In a multi-client environment, clients are constantly requesting data, creating traffic intensity that affects the efficient use of resources on the server. This means that pooling can be very quick to respond to clients, but also causes a lot of cost increases when too much resources are spent on the server [10].

To reduce wasted usage, the servers require the clients to report that they have waiting time for the data (response). That's the background. If the update on the server is fast and a client or several clients are sending requests, it is normal for the reply to be sent. In such cases, the duration of patience is short (usually a few seconds). If several requests return without updating, the client will continue to do so with the new pooling method, increasing the wait time. The client will increase the maximum wait time this time. If the answer is not answered again, the waiting time is marginally increased.

The client will no longer wait until the roof is over. It may return to a faster rate of pooling and in this case some results. Waiting time is reduced if the waiting time waiting for the client starts to return a new request under a certain rate. This model aims to use the resources on the server efficiently in communication with the client.

6. Mobile session security

One of the most important challenges for the client to communicate with the server is the difficulty of continuing the dialogue with multiple HTTP requests. Because the client requests

the server to respond first or nearest when it sends a request to the server. The same is true for mobile device clients.

When the client sends a request to the server, the client receives the incoming response and disconnects from the environment. Then it is natural to send a request back to the server, get the response back, and ask for a fresh one. But this time the question arises as to when the second request should be sent. If the problem is further elicited, it is necessary for the server to remember that the client connected to the client for the duration of the client connection is the same client. This will be resolved if you want to use the IP address. Because the client with the same IP address will be the same client for the server.

In some cases, it is not desirable to use IP. In this case, there is solution. However, the descriptive types will no longer be sufficient for this problem. Using a malicious client IP address or other identifiers, it may send incorrect data to the server, or it may prevent the client from going to the server by entering and hiding messages that they want to send to the server. In such a case, the second question will arise whether the same client is coming.

The solution to the problem is to add session cookies to the first request of the client on the server. In this case, the message will be sent over the HTTPS protocol so that the message will not be read by another client, and the message will be sent to the server as attachment and other information beside the message. In this solution, it is also necessary for the server username and password to be known to the other party. These descriptors may be in the body or any part of the message. The entire message will be encrypted as it will be transmitted via the HTTPS protocol.

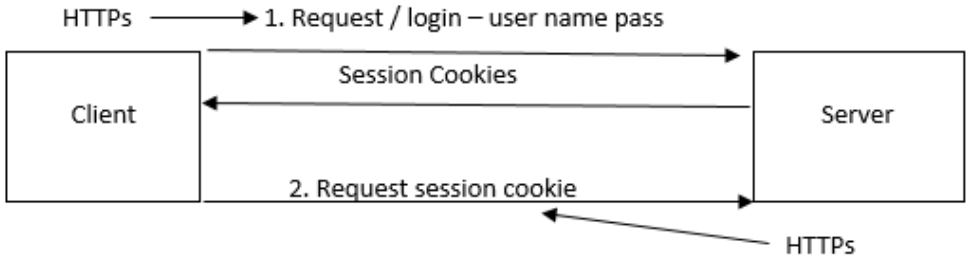

Figure 5. Session cookie usage scheme

The descriptors in the message are enough to verify who the client is. When it is desired to create a session with the client, the server checks the password sent by the client from the database and gives it up to the client. The password is checked to see if the client is the real client.

The server sends back a cookie to know if the cookie (a small amount of data) is a real client in other messages sent by the client. On the second request, the cookie generated by the server is used by the client to be given to the server. This cookie is a piece of information for the client to remember and added to the request when the server is in other requests. On the server side this cookie is matched to the appropriate client. The server finds this cookie as the result of a review of who is the cookie. Because the first request was answered with a small cookie information that was a cookie or session cookie. With descriptive data, the client can tell who the server is. For all subsequent requests, it is understood that the request was sent by the same client.

The same requests are answered by the server as new requests are made, keeping the rights of the same client. Because the user name and password information is received first request, then cookies and cookies are defined by the information.

If the server is not confusing the cookies, it is safe that the client sending the new request is the same client. It is very useful to use the HTTPS protocol in this process.

When connection information (username, password) is sent it needs to be protected. This happens over the HTTPS protocol and in an encrypted form. The fact that the replies are encrypted makes it possible for a session cookie to communicate with the server and allow the server to enter a special client role.

When a session tie is seized in a client (malicious), it is now seen as another client (client being stolen from the client) per this client server. The benefit of using HTTPS is understandable here. Since the data is encrypted in this way, it is encrypted in the cookie and the malicious client cannot fool the server because it cannot open it. Briefly, using HTTPS in communications is essential for secure communication today.

7. Conclusion

Whether open source or closed source, mobile operating systems interpret the concept of cloud differently. But both formats can support communication between them. Different mobile operating systems may reject input when they are communicating with each other, which may create a security breach. For example, they do not communicate with TCP when they communicate with the HTTP protocol. HTTP (S) is the best protocol for communication in the cloud environment.

References

[1] Lenhart, Amanda, "Social Media & Mobile Internet Use Among Teens And Young Adults. Millennials." Pew Internet & American Life Project (2010).

[2] "We Power The Microsoft Cloud", accessed 20 November 2015, http://www.microsoft.com/en-us/server-cloud/cloud-os/global-datacenters.aspx.

[3] "Windows Phone, What Is Cloud?", accesed 15 January 2014, http://www.windowsphone.com/en-us/how-to/wp8/basics/what-is-the-cloud.

[4] Sağıroğlu, Şeref, And Hülya Bulut, "Mobil Ortamlarda Bilgi Ve Haberleşme Güvenliği Üzerine Bir Inceleme." Gazi Üniversitesi Mühendislik-Mimarlık Fakültesi Dergisi 24.3 (2009).

[5] Samantha Morris, "Cloud Types: Private, Public And Hybrid.", Asigra, (2011),

[6] Ben-Ari, "Summarizes The Concurrent Programming Abstraction", Prentice Hall International, (1982).

[7] Ben-Ari, "Summarizes The Concurrent Programming Abstraction", Prentice Hall International, (1982).

[8] Marden, "Why Are Standars Necessary?", "Uses BSC As An Exmple To Show The Need For Both Standard Protocols And A Standard Framework", Communication Network Protocols 2nd Edition, (1986).

[9] “Beal Vangie, “Cloud Computing (The Cloud)”, Webopedia, accessed 12 January 2014, http://www.webopedia.com/term/c/cloud_computing.html.
[10] Kuzu A., Bilgisayar Ağları Ve İletişim, Nobel Yayınları, (2011).
[11] Veugen, Thijs, "A Framework for Secure Computations with Two Non-Colluding Servers And Multiple Clients, Applied To Recommendations." IEEE Transactions on Information Forensics And Security 10.3 (2015): 445-457.