Location Based One-Time Conference Protocol Without Personal Information

Jiyoung Lim(✉) and SangHyun Lim(✉)

Korean Bible University, Seoul, Korea
{jylim, youngmoon92}@bible.ac.kr

Abstract. Nowadays, we produce massive digital data using smartphones and need to share them with others. However, we feel tired of sharing our personal information repeatedly since we live in the society where the excessive personal information is disclosed. We propose a protocol for the location-based one-time communication service using a beacon. In the service, people at the same place could meet at online to share some digital data without revealing their personal information. After meeting, any personal information does not be traceless to each other. People can communicate without disclosure of personal information.

Keywords: Beacon · BLE (Bluetooth Low Energy) signal
One-time communication · Location-based communication

1 Introduction

As smartphones have propagated rapidly, we frequently use digital data as forms of communication. People produce a lot of digital data such as text, picture, video, location information and voice and try to exchange and share them. To do this people should reveal their phone number or ID of SNS application with each other. Sometimes we do not want to share our personal information such as private and public phone numbers and IDs of SNS app. We propose a service protocol that people at the same place could attempt the one-time digital data exchange with each other not to share their personal information.

We choose a beacon among various ways to know people's location information. Our proposal is the offline to online service based on the location information when people working for the same purpose at the same place want to communicate. We use only beacon signal information for the service. Since a beacon sends the signal including location information periodically, people with same beacon signal request the channel for the signal, enter the channel and communicate securely at the same channel. After working, people terminated the service not to store their personal information.

The remainder of this work is organized as follows. Section 2 describes the related work. System architecture and protocol for our proposal are described in Sect. 3. Implementation is described in Sect. 4. In Sect. 5, we conclude our work.

J. H. Park et al. (Eds.): PDCAT 2018, CCIS 931, pp. 50–55, 2019.
https://doi.org/10.1007/978-981-13-5907-1_5

2 Related Works

2.1 Beacon

Beacon uses the Bluetooth 4.0, Bluetooth Low Energy (BLE) technology [1]. While former Bluetooth technology provides in pairing between two devices, a beacon broadcasts BLE signals periodically and continuously. Devices such as smart phones receive signals, send the request with the signal information to the server through wireless networks such as LTE and WIFI. and get a response from a server.

A beacon sends a fixed BLE signals with a small amount of data such as the location information. iBeacon provides the location information through UUID, Major and Minor which can be assigned as values determined randomly by the service provider [2]. A service provider can divide a place into several blocks and determine the value of UUID, major, and Minor for each block. Table 1 shows an example of UUID, Major and Minor. A beacon product provides the functions of the iBeacon and EddyStone, since iBeacon of Apple and EddyStone of Google [3] are considered as the Beacon standard.

Table 1. An example of UUID, major and minor

Korean Bible Univ.		Bldg #1	Bldg #2	Bldg #3
UUID		123A4B3-FC18-1234-9F12-C90DB1F207F1		
Major		1000	2000	3000
Minor	Room #1	101	101	101
	Room #2	102	102	102
	Room #3	103	103	103

3 Our Proposed Protocol

Our protocol aims to communicate with each other through the secure channel not to exchange their own personal information. We describe the initial phase, the channel connection phase and the beacon and session management phase.

3.1 Initial Phase

First of all, if a device runs the client app, it sends the initial message to the server using HTTPS rest to request the port for SSL socket and the token for the one-time session maintenance. The server responses the client's request, allocates the resource and wait the client's SSL TCP socket connect request. At the initial phase, completes that the client tries to connect the SSL socket and the server accepts it. Figure 1 shows the initial procedure.

The initial phase restarts when the client that has used the channel sends another new beacon signal asking to connect a new channel corresponding the signal. In the protocol, we do not keep any client's personal information and need only the token to maintain the current session. Token means that the client belongs to a special group at the same place and is one of means that determining the session life time.

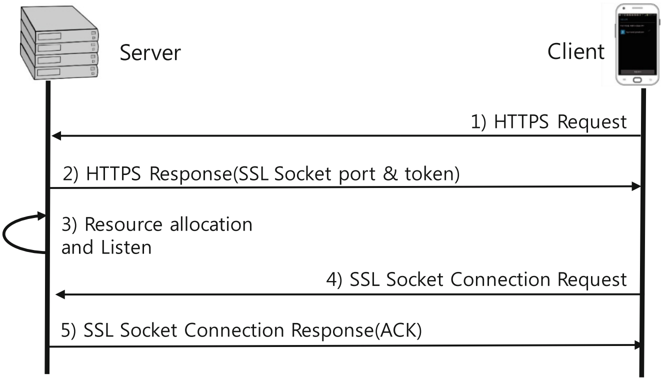

Server
Client
1) HTTPS Request
2) HTTPS Response(SSL Socket port & token)
3) Resource allocation
and Listen
4) SSL Socket Connection Request
5) SSL Socket Connection Response(ACK)

Fig. 1. Initial procedure between server and client

3.2 Channel Connection Phase

Because the beacon signal includes their position information, people in the same room or block have the same beacon signal. If they send the same signal information to the server, they communicate with the same channel through the SSL socket.

If a device receives the beacon signal, it sends the beacon signal information to the server through the SSL socket connected in the initial phase. The server checks if a channel for the beacon signal information is allocated or not. If it does not exist, the server creates a channel for the beacon signal information. After verifying the existence of the valid channel, the device is added to the channel user list. Hence forward, people with same beacon signal can transmit and receive the data among themselves through the SSL socket. Figure 2 shows the connection procedure.

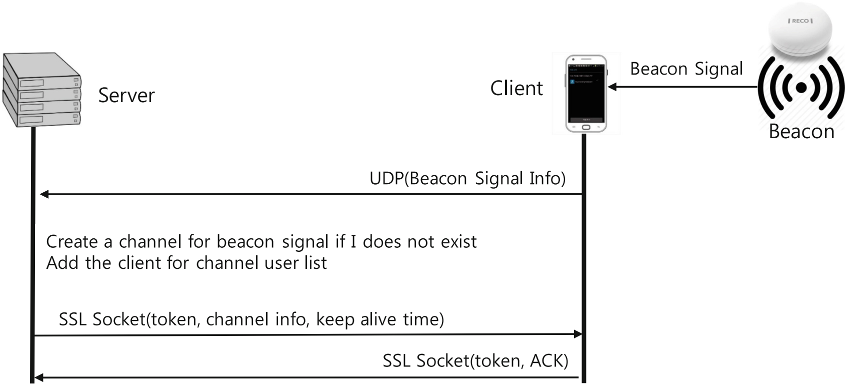

RECO
Server
Client
Beacon Signal
Beacon
UDP(Beacon Signal Info)
Create a channel for beacon signal if I does not exist
Add the client for channel user list
SSL Socket(token, channel info, keep alive time)
SSL Socket(token, ACK)

Fig. 2. Channel connection procedure

3.3 Beacon Signal and Session Management

Periodic beacon signal reception is important to maintain the channel connection. Devices receiving beacon signals should send the signal information to the server through UDP whenever the beacon signal arrives. Because the beacon signal information could not arrive at the server according to UDP property, the server checks the channel status, extend the channel session life time, inform the client of keep alive time and response ACK every 30 s through SSL sockets. When the server does not receive it, the server request the client tolerates during 120 s. After timer expires, the server decides that the client leaves the place, disconnects the channel from the device and nullify the token of the client. Figure 3 shows the beacon signal and session management procedure.

The client wants to disconnect the channel by sending a terminating message to the server. The server releases the channel and nullify the token. Hence, if the client wants to connect the server again, the server does not store the client's information at the repository such as database and should issue the token newly.

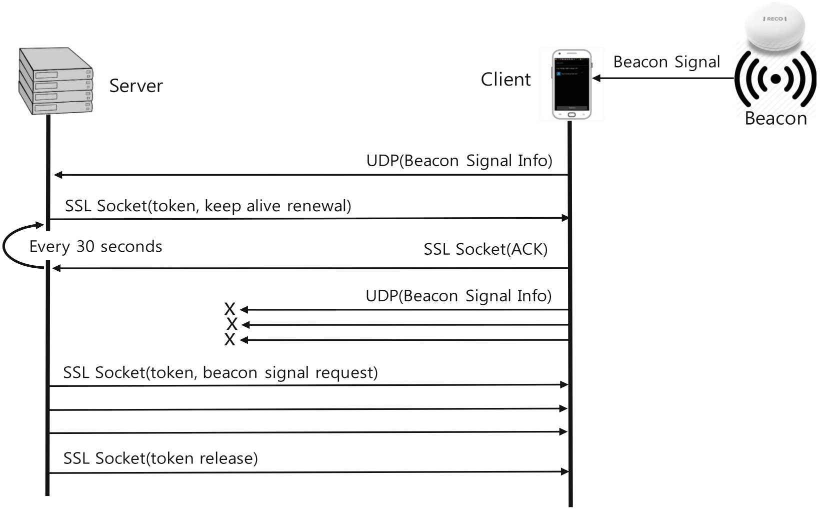


Fig. 3. Beacon signal and session management procedure

4 Implementation

4.1 Node.js

Node.js is a server middleware based on the google chrome V8 java script engine providing the asynchronous I/O. It is a high performance server development technology based on a single thread. Global service provider such as PayPal and Groupon changed their service platform based on Node.js.

To process I/O such as file and network I/O, a program sends I/O requests to Node.js and processes other tasks until arriving the event of I/O completion in Fig. 4. Because CPU time is used only for processing without waiting, Node.js increases the utilization and performance. Since only a single thread processes several requests, Node.js is considered as one of a solution for C10k problem [4].

Node.js always does not shows the higher performance than that of java and others. In case of low CPU processing and a lot of concurrent connection processing, Node.js is superior to others.

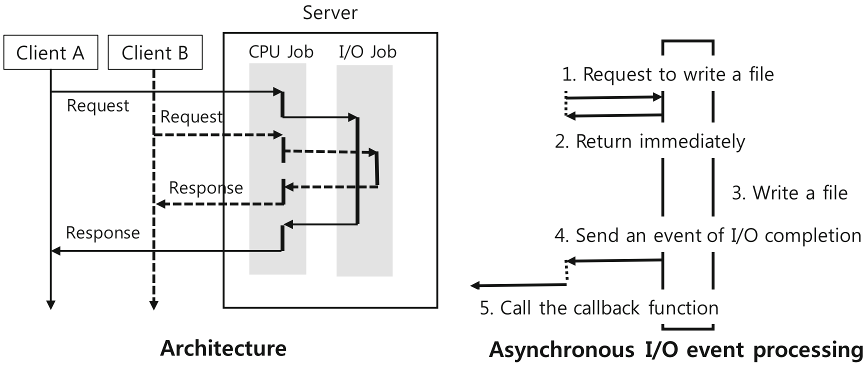
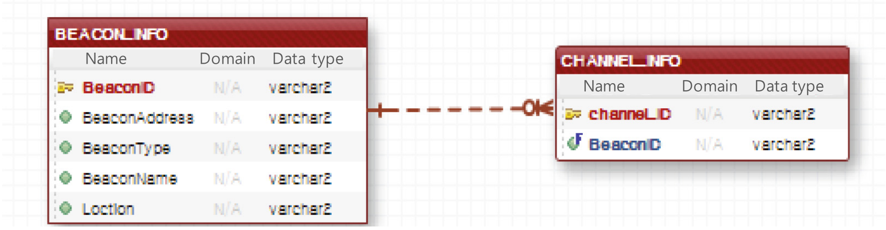

Server
Client A
Client B
CPU Job
I/O Job
Request
Request
Response
Response
Architecture
1. Request to write a file
2. Return immediately
3. Write a file
4. Send an event of I/O completion
5. Call the callback function
Asynchronous I/O event processing

Fig. 4. Architecture and event processing of Node.js

4.2 Server and Client

Server based on Cent OS 7 developed server backend and front end using Node.js. In Fig. 5 shows two database tables of oracle 11 g local DBMS concerning beacons and channels. The client is developed using Android SDK. The name of client app is Here. O meaning same place and one-time online. We use a beacon made by RECO [5] and BLE API provided by Android. Figure 6 shows the splash screen image of app that waits the HTTPS response for the server. The success of HTTPS connection in splash goes to next screen. If button 'Enter' in the main layout image is clicked, beacon signal information receiving from a beacon is sent to a server and the channel is connected for the beacon. Then people with same beacon signal can communicate with each other.



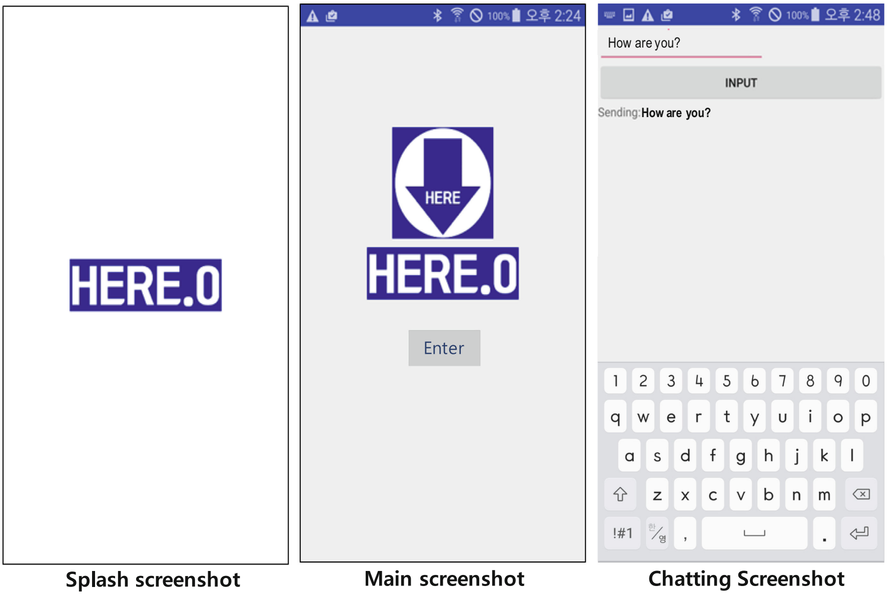
BEACON_INFO

Name	Domain	Data type
BeaconID	N/A	varchar2
BeaconAddress	N/A	varchar2
BeaconType	N/A	varchar2
BeaconName	N/A	varchar2
Loction	N/A	varchar2

CHANNEL_INFO

Name	Domain	Data type
channel_ID	N/A	varchar2
BeaconID	N/A	varchar2

Fig. 5. Beacon and channel database table



How are you?

Sending: How are you?

Splash screenshot

Main screenshot

Chatting Screenshot

Fig. 6. Client screenshots

5 Conclusion

We proposed a secure protocol for the location-based one-time communication service using a beacon without fear of disclosure for personal information. The session was maintained by the secure SSL socket and one-time token based on beacon signal and smartphone mac address and another beacon signal information through UDP. Digital data did not be encrypted however additional encryption is not difficult. We thought our protocol is useful to people that want the conference without exchange of personal information.

References

1. Kalliola, K., High accuracy indoor positioning based on BLE. Nokia research center presentation (2011)
2. Gast, M.S.: Applications with IBeacon: Proximity and Location Services with Bluetooth Low Energy. O'Reilly Media Inc., Sebastopol (2014)
3. Dasgupta, A., Nagaraj, R., Nagamani, K.: An internet of things platform with Google eddystone beacons. J. Softw. Eng. Appl. **9**(06), 291 (2016)
4. The C10K problem. http://www.kegel.com/c10k.html
5. RECO Beacon. http://reco2.me/reco-beacon/