Contact Tracing

Bluetooth Specification

Preliminary - Subject to Modification and Extension

April 2020

v1.1

Contents

Overview

This document provides the detailed technical specification for a new privacy-preserving Bluetooth protocol to support Contact Tracing. Contact Tracing makes it possible to combat the spread of the COVID-19 virus by alerting participants of possible exposure to someone who they have recently been in contact with, and who has subsequently been positively diagnosed as having the virus. The Contact Detection Service is the vehicle for implementing contact tracing and uses the Bluetooth LE (Low Energy) for proximity detection of nearby smartphones, and for the data exchange mechanism.

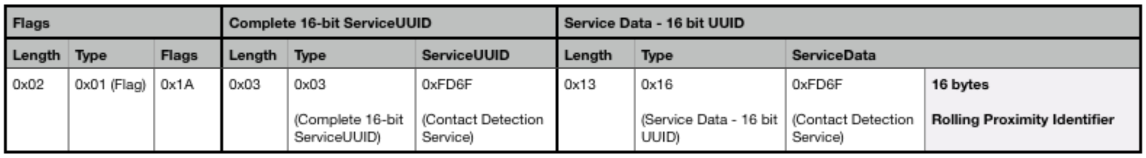
Definitions

- Contact Detection Service - The BLE service for detecting device proximity.
- Tracing Key - A key that is generated once per device.
- Daily Tracing Key - A key derived from the Tracing Key every 24 hours for privacy consideration.
- Diagnosis Key - The subset of Daily Tracing Keys which are uploaded when the device owner is diagnosed positive for the COVID-19 virus.
- Rolling Proximity Identifier - A privacy preserving identifier derived from the Daily Tracing Key and sent in the bluetooth advertisements. It changes every ~15 minutes to prevent wireless tracking of the device.

Contact Detection Service

Contact Detection is a BLE service registered with the Bluetooth SIG with 16-bit UUID 0xFD6F, it is designed to enable proximity sensing of Rolling Proximity Identifier between devices for the purpose of computing an exposure event.

Devices advertise and scan for the Contact Detection Service by way of its 16-bit service UUID. The Service Data type with this service UUID shall contain a 128-bit Rolling Proximity Identifier that changes periodically.



Flags			Complete 16-bit ServiceUUID			Service Data - 16 bit UUID			
Length	Type	Flags	Length	Type	ServiceUUID	Length	Type	ServiceData	
0x02	0x01 (Flag)	0x1A	0x03	0x03 (Complete 16-bit ServiceUUID)	0xFD6F (Contact Detection Service)	0x13	0x16 (Service Data - 16 bit UUID)	0xFD6F (Contact Detection Service)	**16 bytes** **Rolling Proximity Identifier**

Advertisement Payload

The Contact Detection Service payload shall be ordered as shown below and shall not include other data types.

The Contact Detection Service payload has three sections:

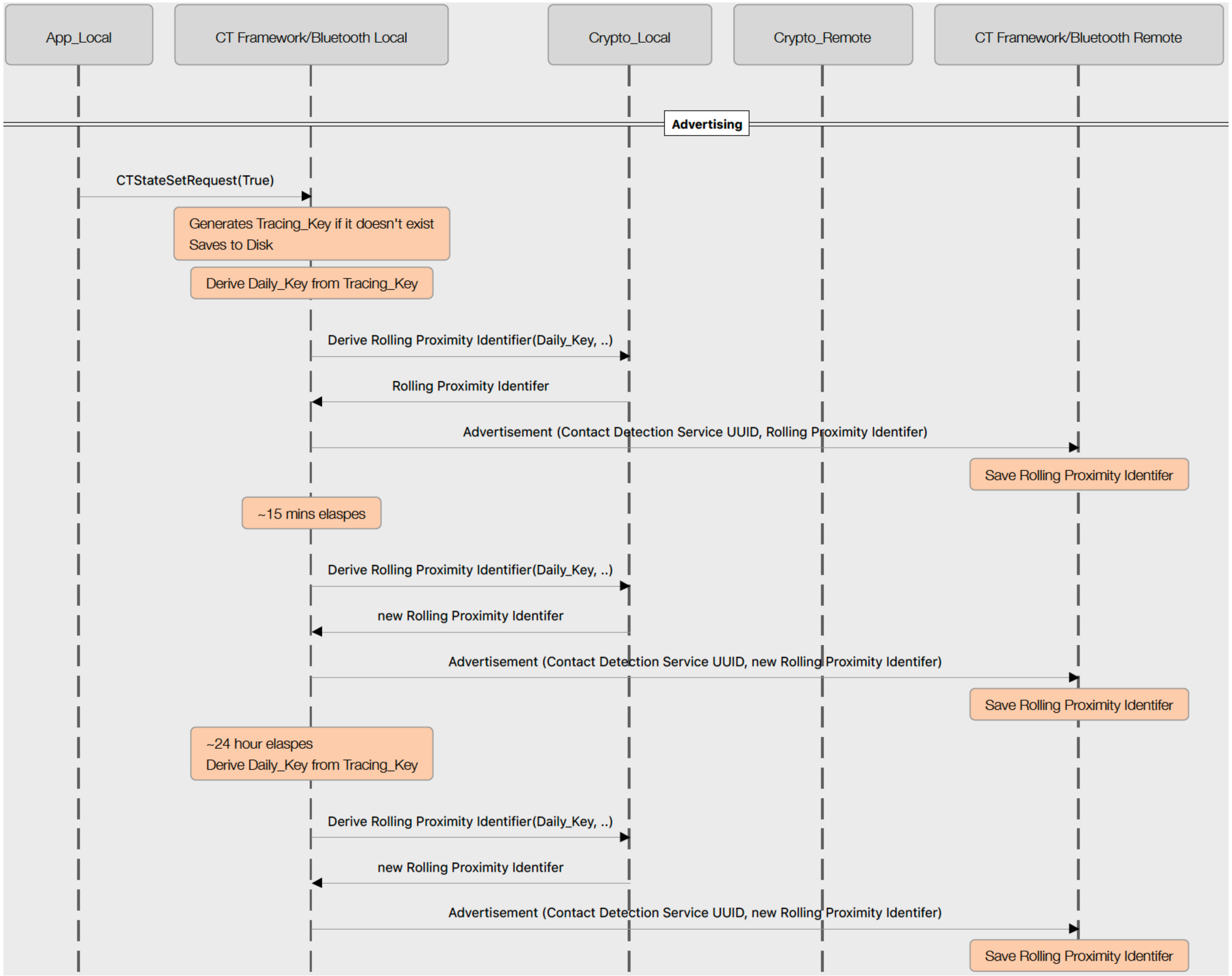
1. Flags Section: LE general discoverable mode (bit 1) shall be set to 1.
2. Complete 16-bit Service UUID Section: the UUID is 0xFD6F, and shall precede the Service Data section.
3. Service Data 16-bit UUID Section: the content of service data payload shall be a 128-bit Rolling Proximity Identifier.

Advertising Behavior

- Advertisements are to be non-connectable undirected of type `ADV_NONCONN_IND` (Section 2.3.1.3 of 5.2 Core Spec).
- The advertiser address type should be Random Non-resolvable.
- On the platforms supporting Bluetooth Random Private Address with randomized rotation timeout interval, the advertiser address rotation period shall be a random value, greater than 10 minutes and less than 20 minutes.
- The advertiser address and Rolling Proximity Identifier shall be changed synchronously so address and Rolling Proximity Identifier can not be linked.
- A separate advertising instance should be used, if HW allows, to provide advertising reliability and flexibility in choosing optimal interval.
- The advertising interval is subject to change, but is currently recommended to be 200-270 milliseconds.

Advertising Flow

The following diagram illustrates the flow of advertisement between devices.


App_Local
CT Framework/Bluetooth Local
Crypto_Local
Crypto_Remote
CT Framework/Bluetooth Remote
Advertising
CTStateSetRequest(True)
Generates Tracing_Key if it doesn't exist
Saves to Disk
Derive Daily_Key from Tracing_Key
Derive Rolling Proximity Identifier(Daily_Key, ..)
Rolling Proximity Identifer
Advertisement (Contact Detection Service UUID, Rolling Proximity Identifer)
Save Rolling Proximity Identifer
~15 mins elaspes
Derive Rolling Proximity Identifier(Daily_Key, ..)
new Rolling Proximity Identifer
Advertisement (Contact Detection Service UUID, new Rolling Proximity Identifer)
Save Rolling Proximity Identifer
~24 hour elaspes
Derive Daily_Key from Tracing_Key
Derive Rolling Proximity Identifier(Daily_Key, ..)
new Rolling Proximity Identifer
Advertisement (Contact Detection Service UUID, new Rolling Proximity Identifer)
Save Rolling Proximity Identifer

Scanning Behavior

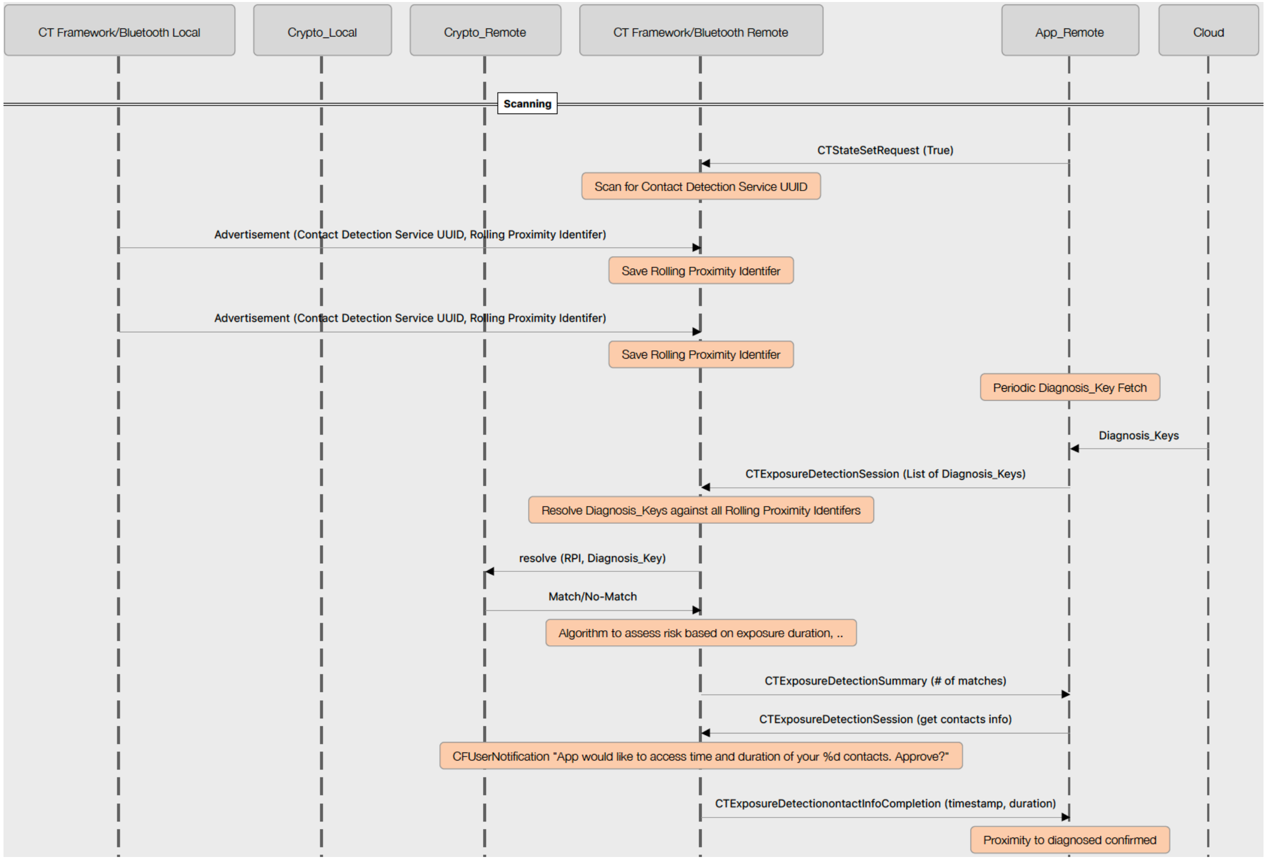
- Discovered Contact Detection Service advertisements shall be kept on the device.
- Scan results shall be timestamped and RSSI-captured per advertisement.
- Scanning interval and window shall have sufficient coverage to discover nearby Contact Detection Service advertisers within 5 mins.
- Scanning strategy that works best is opportunistic (leveraging existing wakes and scan windows) and with minimum periodic sampling every 5 mins.

Scan Power Considerations

- Platforms running the Contact Detection Service should be designed to account for large volume of advertisers in public spaces that shall rotate their Random Non-resolvable address and Rolling Proximity Identifier frequently.
- Wherever supported by HW and OS, Bluetooth controller duplicate filters and other HW filters should be used to prevent excessive power drain.

Scanning Flow

The following diagram illustrates the flow and behavior of device scanning.


CT Framework/Bluetooth Local
Crypto_Local
Crypto_Remote
CT Framework/Bluetooth Remote
App_Remote
Cloud
Scanning
CTStateSetRequest (True)
Scan for Contact Detection Service UUID
Advertisement (Contact Detection Service UUID, Rolling Proximity Identifer)
Save Rolling Proximity Identifer
Advertisement (Contact Detection Service UUID, Rolling Proximity Identifer)
Save Rolling Proximity Identifer
Periodic Diagnosis_Key Fetch
Diagnosis_Keys
CTExposureDetectionSession (List of Diagnosis_Keys)
Resolve Diagnosis_Keys against all Rolling Proximity Identifers
resolve (RPI, Diagnosis_Key)
Match/No-Match
Algorithm to assess risk based on exposure duration, ..
CTExposureDetectionSummary (# of matches)
CTExposureDetectionSession (get contacts info)
CFUserNotification "App would like to access time and duration of your %d contacts. Approve?"
CTExposureDetectionontactInfoCompletion (timestamp, duration)
Proximity to diagnosed confirmed

Privacy

Maintaining user privacy is an essential requirement in the design of this specification. The protocol does this by the following means:

- The Contact Tracing Bluetooth Specification does not use location for proximity detection. It strictly uses Bluetooth beaconing to detect proximity.
- A user's Rolling Proximity Identifiers changes on average every 15 minutes, and needs the Daily Tracing Key to be correlated to the user. This reduces the risk of privacy loss from advertising them.
- Proximity identifiers obtained from other devices are processed exclusively on device.
- Users decide whether to contribute to contact tracing.
- If diagnosed with COVID-19, users consent to sharing Diagnosis Keys with the server.
- Users have transparency into their participation in contact tracing.