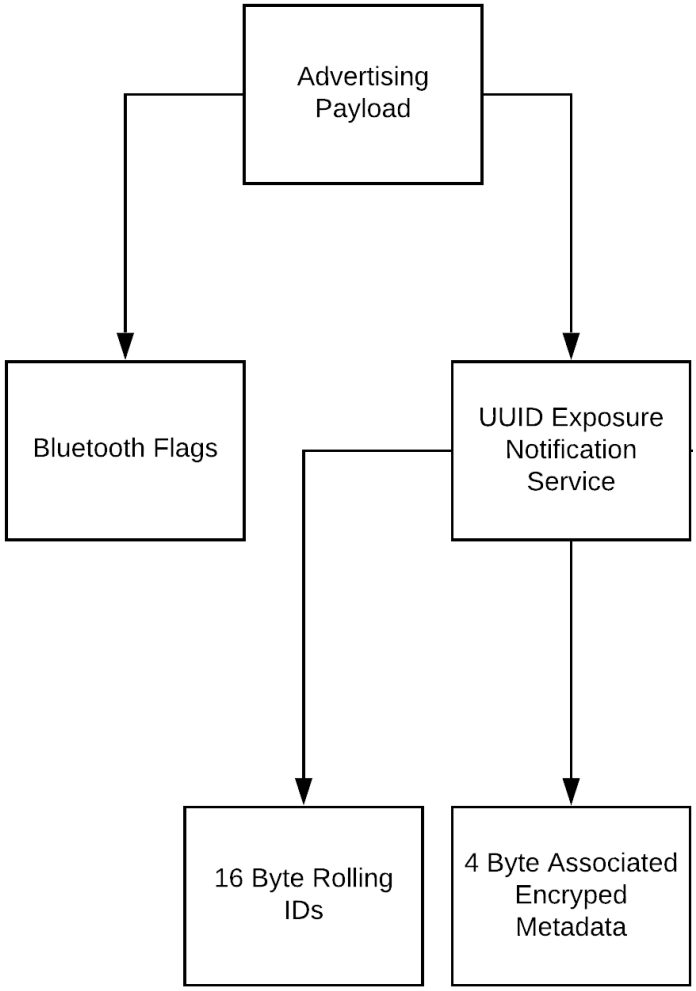
Weekly Blog 2

1. Understood about GAEN (Google/Apple Exposure Notification) Advertising Payload



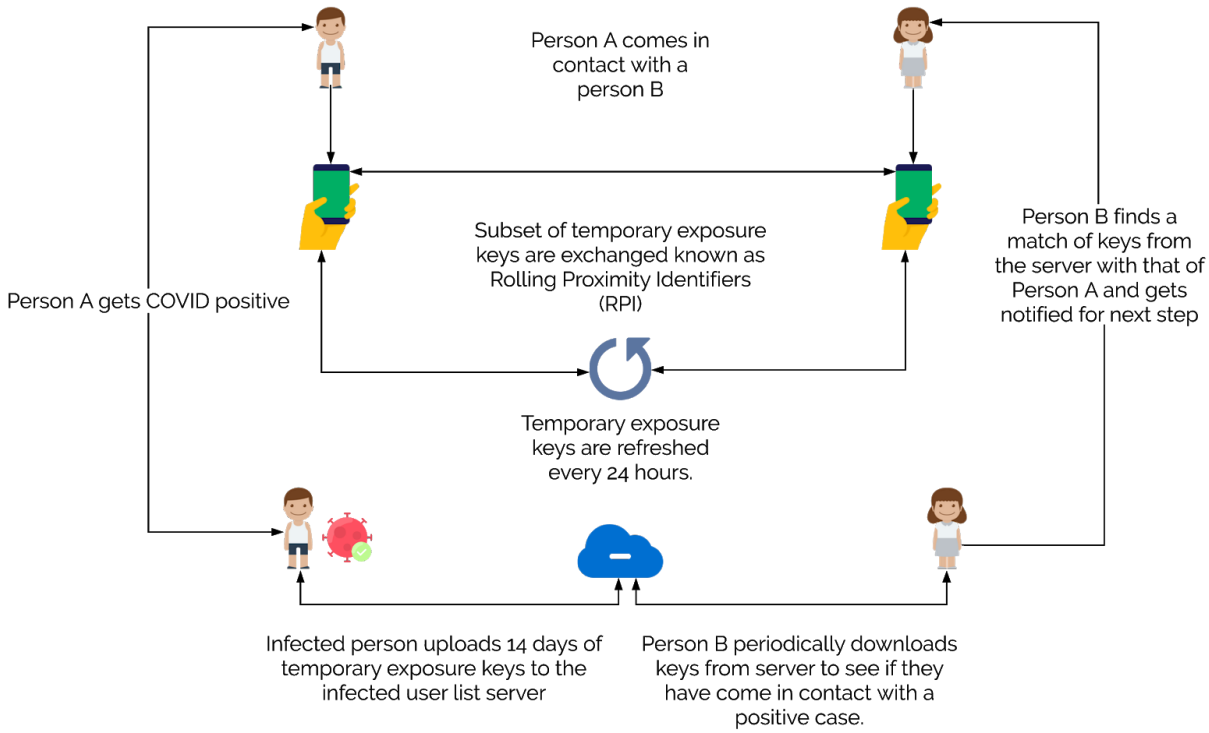
2. Understood Flags Section:
    a. Bluetooth Low Energy general discoverable mode (bit 1) shall be set to 1.
3. Understood UUID:
    a. A universally unique identifier is a 128-bit number.
4. Understood about the 16-byte rolling proximity identifier, which is refreshed every 15 minutes to prevent user tracking.

   a. This is sent as part of the advertising payload, which is made up of other components.
5. Understood about the Associated Encrypted Metadata (AEM). This data is of 4 bytes, which is a part of the advertising payload.
   a. Byte 0 - Versioning.
      i. Bits 7:6 - Major version (01).
      ii. Bits 5:4 - Minor version (00).
      iii. Bits 3:0 - Reserved for future use.
   b. Byte 1 - Transmit power level.
      i. This is the measured radiated transmit power of Bluetooth Advertisement packets and is used to improve distance approximation. The range of this field shall be -127 to +127 dBm. (how strong the Bluetooth signal might be at a known distance)
   c. Byte 2 - Reserved for future use.
   d. Byte 3 - Reserved for future use.
6. Understood Temporary Exposure Key
   a. Generated every 24 hours known as EKRollingPeriod which is set as 24 hours in the current version of GAEN
7. Understood Diagnosis Key:
   a. The subset of Temporary Exposure Keys uploaded when the device owner is diagnosed as positive for the coronavirus.

How does the detection process work?

Each set of IDs and the Metadata received over BLE is stored and later periodically checked to see if they have come in contact with a COVID positive case.

1. When an infected person takes a COVID-19 (positive) test, an operator of an EN backend, the system assigns a one-time code to the infected person, which allows the infected person to upload its most recent (typically representing the last 14 days) Temporary Exposure Keys to the backend server. Temporary Exposure Keys uploaded by an infected person are referred to as Diagnosis Keys.
2. End-user devices periodically download Diagnosis Keys from the EN backend.
3. Devices compute exposure risks locally, and EN applications inform the user of the next steps to take.


Person A comes in contact with a person B
Subset of temporary exposure keys are exchanged known as Rolling Proximity Identifiers (RPI)
Person A gets COVID positive
Person B finds a match of keys from the server with that of Person A and gets notified for next step
Temporary exposure keys are refreshed every 24 hours.
Infected person uploads 14 days of temporary exposure keys to the infected user list server
Person B periodically downloads keys from server to see if they have come in contact with a positive case.

Contact Tracing App Privacy: What Data Is Shared By Europe's GAEN Contact Tracing Apps

What research question does the paper address?

Aim: The primary aim of the paper is to research what kind of data is being sent over using the GAEN API. More specifically, it explains the data sources concerning the GAEN API. However, we will be focussing more on the data sources and the possible issues related to it, rather than specific data elements GAEN shares. This is more useful for our project implementation as we could steer clear from the potential areas that could do harm to privacy.

What does it mainly discuss about?

1. The initial introduction to the paper mainly discusses the security and privacy-related issues in some of the COVID contact tracing apps in Europe, although this could be applicable in general as well. Some of the main issues that could be addressed while implementing as a team would be as follows:
    a. Reduce the number of third parties handling the user data. For instance, an app in Poland used Google's Firebase for handling app configuration settings. Even though the paper does not explain the issues related to it. Some of the main issues are:
        i. The first and foremost issue is the data is entirely hosted on servers that we do not own. It is obvious that identity sensitive information requires the utmost protection. As such, limiting the number of parties handling the data is vital.
        ii. As evident from public sources, the number of queries for querying the data is also limited.
2. There is an immense focus on Google Play Services connecting with Google every 20 minutes for linking multiple devices together. For this IP address based location data are used for the purpose. The paper explicitly mentions that it is not aware of the types of IP addresses Google uses for the above-mentioned purpose. As such, being one of the world's most trusted companies, it is obvious about the security concerns Google would undertake regarding what kind of data it accesses. Furthermore, Google and Apple claim to not have access to it.
    a. The paper also suggests restricting unwanted google play services which could reduce data being sent to Google.
    b. Firewall apps can be installed to prevent unnecessary google play service connections
3. Since all of these apps are being developed in warp speed, some of the privacy concerns are inevitable. Some of these claims are and can be accepted as infringement or intrusion to the user's data.

What was the conclusion of the research?

The conclusion of the research is how we can limit the data sent by Google Play services by disabling most of its sub-services like Google Maps, Movies and Books etc. These data included IMEI, hardware serial number, SIM serial number, handset phone number and user email address with Google. Even though the data concerns can only be limited by limiting many of the features, it cannot be completely prevented. However, owing to its many advantages like the BLE broadcast model, decentralized approach and whatnot, it seems to be still better than the available apps on the market

How can you apply this knowledge to your own project?

1. The most useful piece of information that we understand from the paper is the prevention of data usage by multiple third parties. Specifically:
    a. Avoiding handling of data by more number of third parties.
    b. Limiting google play service data sending.
    c. Use of firewalls.
    d. Avoiding the collection of user data for sign up purposes. For instance, COVIDSafe requires you to sign up.

Adding Location and Global context to the Google/Apple Exposure Notification Bluetooth API

What research question does the paper address?

Aim: The aim of the paper is to include location and global context to the GAEN API. The authors argue to enhance the information that is given to users through the exposure notification service through the use of location and global context. For instance,

1. It will help users to self-assess their exposure in terms of the specific circumstances they were exposed to.
2. The number of false-positives could be ruled out significantly. For example, an app might report a contact with a COVID positive case even if the two cases were separated by a wall and not technically exposed to each other.
3. Help Public Health Authorities to make more informed decisions.

The primary challenge to the approach is the use of the location-based analysis which is against GAEN's compliance policies. As such, the authors suggest four methodologies based on GAEN to include global context and location in a way it does not lead to privacy concerns.

1. GPS logs stay on-device app, data does not leave the phone, and cannot be visualized:

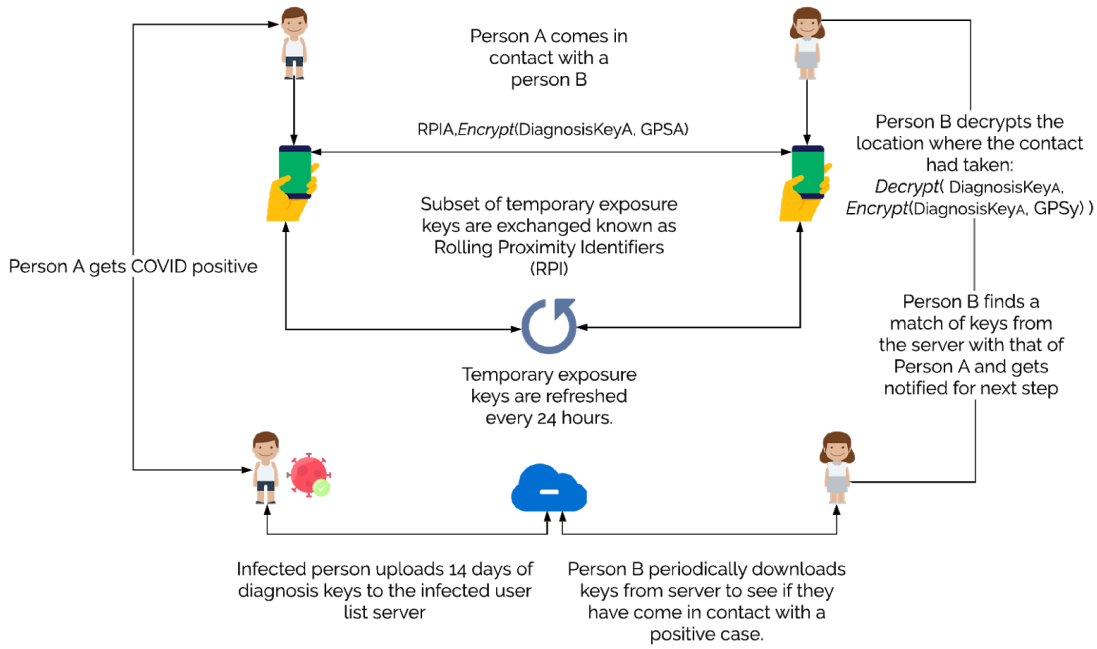
   In this approach, the authors suggest the tagging of Rolling Proximity Identifiers (RPI) with location when a user is notified of exposure notification. Specifically, the user's location is tagged with RPI only when there is an encounter with a COVID positive case.
   a. The use of the approach is that the location data is stored locally and not available to the public.
   b. The misuse of this approach is non-compliant with GAEN policies as it explicitly uses location-based data.
2. BLE for proximity, GPS for context, GPS is blurred for privacy, data does not leave the phone, and cannot be visualized

   In this approach, the location is not pinpointed and stored on the device. Instead, it is blurred based on the local population density. For instance, if a user comes in contact with a COVID positive case, the exposure notification would notify the user that he/she had come in contact with an infected case within 200m of the location in which they were in previously. The user could use this to assess and remember if they have been exposed.
   a. Location is not available to even hackers as it is approximated.
   b. Data stored locally.
   c. The possible downside is, context-related information accuracy is reduced.
3. GPS in existing GAEN protocol

   In this approach, the authors suggest the modification of the advertising payload

structure very minimally, with the inclusion of the GPS data as well. The GAEN protocol as of now advertises its payload consisting of UUID, RPI, and AEM. The authors suggest the encryption of GPS data with the diagnosis keys.

The flow chart for this is as follows:



What was the conclusion of the research?

The conclusion of the research is that the authors believe the inclusion of location and global context-based data could prove to be more accurate and useful for users and public health authorities. They believe the 4th approach cited above conforms to the GAEN guidelines by including encrypted GPS data as part of the advertising payload.

How can you apply this knowledge to your own project?

As of now, the authors have assumed that the 4th approach could be a useful approach. There is no empirical evidence given in the paper that backs their theory. This could be considered only if the project is taken for further research and scaling. Right now, we will stick to the documented and already available GAEN API. This could be considered for scrutinizing security towards the later stages of the project.

On using Bluetooth-Low-Energy for contact tracing

What research question does the paper address?

Aim: The paper mainly describes the technical specifications of a Bluetooth Advertising Payload and the type of models used.

The payload is made up of:

- Advertising address: 48 bits representing the address of the device
- Advertising data: 31 bytes long, mainly used for broadcasting packets.
- Advertising interval: Period over which the packets are broadcasted.
- scanInterval: which determine the time between two consecutive scans, and the
- scanWindow: which determine the duration of a scan

Three types of models:

1. Broadcast model
2. Connected model
3. Hybrid model

Broadcast model: Contact tracing has adopted this model, by sending 20 bytes of data in the advertising data element. Specifically, the 16 byte RPI and the 4-byte AEM is sent in the packet.

Connected model: While, the broadcast model does not rely on any connection between two devices, the connected model relies on it for transferring the data packets between the devices. The main drawback to the approach is the transfer of data packets happens only for the duration the connection persists.

Hybrid model: In this hybrid approach, for devices without the capability of broadcasting packets without a connection, the connected model is used. Whenever the broadcasting capability exists, the broadcast model is used.

The connection model is mainly used in situations where you need an acknowledgement from the recipient device that the packet has been received. Since contact tracing is mainly focused on scalability, the broadcast model is more suited as it is less complicated and requires no connections.

Since GAEN is primarily using the broadcast mode, we would be focusing on that. The broadcast model can be extended using the following extensions. These extensions can be used to include more data into the contact tracing advertising data section. However, this is only supported by the latest android and iOS versions. These extensions are only needed for advanced Bluetooth protocols, which are not explicitly mentioned in the paper as well. The scope of these extensions needs to be explored further.

What was the conclusion of the research?

The conclusion of the research is the exploration of different modes of Bluetooth: Broadcast, connected, and hybrid modes. The authors mainly lean towards Broadcast models due to their advantage of scalability. Furthermore, it does not require any connection with other devices to send and receive packets. In the future, if any enhancements are required, the extensions to the broadcast model can be used.

How can you apply this knowledge to your own project?

The knowledge gained from the paper is an in-depth analysis of GAEN advertising payload. Leveraging this, I could really understand how the packets are transmitted between two devices by broadcasting and how the COVID contact tracing uses this. Furthermore, the structure of the Bluetooth payload could also be understood.

References:

1. Leith, D.J., and Farrell, S., 2020. Contact Tracing App Privacy: What Data Is Shared By Europe's GAEN Contact Tracing Apps.
2. Raskar, R., Singh, A., Zimmerman, S., and Kanaparti, S., 2020. Adding Location and other Context to the Google/Apple Exposure Notification Bluetooth API. arXiv preprint arXiv:2007.02317.
3. Mathieu Cunche, Antoine Boutet, Claude Castelluccia, Cédric Lauradoux, Daniel Le Métayer, et al. On using Bluetooth-Low-Energy for contact tracing. [Research Report] Inria Grenoble Rhône-Alpes;INSA de Lyon. 2020. ffhal-02878346v4f
4. https://covid19-static.cdn-apple.com/applications/covid19/current/static/contact-tracing/pdf/ExposureNotification-BluetoothSpecificationv1.2.pdf