Asynchronous Contact Tracing, Fighting Pandemics with Internet of Things

Abdul Qadir Khan

Université Cote d'Azur, Master 2 Informatique (UBINET)

Internship Report

Asynchronous Contact Tracing, Fighting Pandemics with Internet of Things

Student

Abdul Qadir Khan

Supervisor

Luigi Liquori (H.d.R., Ph.D. INRIA Research Director, EPC KAIROS)

Academic Supervisor

Prof. Yves Roudier (UCA, SPARK/I3S)

Organization

Inria (KAIROS)-Sophia Antipolis Méditerranée

Duration

15 May, 2021 - 15 September, 2021

September 13, 2021

Acknowledgements

First, I would like to express my sincere gratitude to my internship advisor Luigi Liquori for his guidance and supervision throughout the internship which helped me to achieve this goal. Also I would like to thank my pedagogical supervisor Prof. Yves Roudier, Head of Master informatique, for his interest in my work and to Prof. Guillaume Urvoy-Keller, Head of Master (UBINET). Also, my gratitude goes to all UBINET/CASPAR Professors.

My sincere thanks also goes to Telecom Italia Mobile, especially Dr. Enrico Scarrone and Dr. Piero Cena, for their valuable time and resource for this internship. I would also like to pass my sincere gratitude to all the ETSI SmartM2M Technical Committee who participated in the process of the standardization and give me the opportunity to participate in the making of standard. It will be insincere if I do not thank the whole Kairos INRIA/I3S/CNRS team, they welcomed me warmly and helped me throughout my internship.

I would like express my very profound gratitude to my parents, family and friends for supporting me and encouraging me throughout my study.

Finally, I would like to thank INRIA Sophia Antipolis - Méditerranée, Université Cote d'Azur and the Computer Science Department at COMSATS University Islamabad, Pakistan for selecting me as an Exchange student and giving me opportunity to gain knowledge and experience in one of the best university and research institutes in France.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In December 2019 the world saw a rise of new virus from a part of world and spread in all parts of the world in just three to four months. The virus was named SARS-CoV-2. The main weapon against this virus is social distancing which results in lock downs in the cluster. As the world saw a long lockdown between March to September 2020, which had severe impact on economy and the personal life of humans. By the time when there are no vaccines available, the other relevant tool in the fight against the virus is testing. Unfortunately, widespread testing of large populations in a very short time is practically not possible. For example for a country of hundred million population will require a huge amount of time and resources to do so. This is clearly an unworkable solution to the problems raised by the current pandemic and is common to most countries across the world. This also to be noted that many people are unwilling to do COVID-19 tested for social reasons, such as job restriction, economic consequences, violation of private life, or even fear of quarantine. It is now well understood that without widespread testing of the population, the only weapon against COVID-19 is lockdown and subsequent severe economic and social disruption.

There was a method used in Europe back in 16th century to trace the people called Contact Tracing. The principles remain the same today whether carried out by phone, mail or personal contact. The aim is to identify the origin of the infection and to where, or to whom, it has been transferred. If receipt of this information is followed immediately by isolation, treatment and aggressive decontamination, it can lead to containment and the gradual elimination of the disease itself.

Digital Contact Tracing idea was initially discussed in 2007. Digital Contact Tracing is tracing user using Smart mobile phones. The possible features of the phone that can help in the tracing are Bluetooth and GPS. In 2018 a patent application from Facebook

discussed the proximity and trust model using Bluetooth. There are different methods of Digital Contact Tracing.

- **Bluetooth proximity tracing:** By using Bluetooth and specially Bluetooth low energy protocol to trace the user in communication range. The users phones share some random pseudonym that are used in tracing. Bluetooth cryptographic techniques make it more private and secure medium of tracing. Also it require less power usage as compared to GPS mechanism.
- **Location tracking:** This mechanism uses the GPS feature of the smartphone. This type of the tools are more private then the Bluetooth mechanism.
- **GEO-QR code tagging:** This method use QR-code, where user scan the QR code using their smartphones and register their visit to a public place like restaurant, hotel, museum etc. In this method the user have the control of their privacy they voluntarily scan the code without downloading an application on phones. Later if the case is detected all the register contacts are informed about the virus infection.
- **CCTV with facial recognition:** facial recognition can also be used for this purpose. This system may or may not store the user information in the database.

1.1 Context of Internship

This internship is a Research and Development Internship which is in collaboration with the standardization bodies of European Telecommunication Standard Institute (ETSI). The Kairos INRIA/I3S/CNRS team participates to the ETSI SmartM2M Technical Committee working on protocol designed to fight against pandemics named "Asynchronous Contact Tracing" [1].

The main objective of the internship was to take part in designing a protocol and contribute to Standardization with ETSI and to implement the protocol with the help of ETSI-oneM2M based middleware implementation platform ICON provided by Telecom Italia Mobile (TIM).

oneM2M is an ETSI standard [2] developed for IoT system focus on developing common service layer that can be readily embedded with in software and hardware to connect the devices with M2M servers.

1.2 About Organization

The Inria Sophia Antipolis-Méditerranée Research Center was established in 1983. Inria is the world known research laboratory driving digital research and innovation in France for more than 50 years. There are almost 35 research groups doing research in different fields of science. Inria have different campuses around France. The environment inside Inria is so diverse, working environment is very friendly which is helpful for students to learn in productive way. Kairos is a research team based in Inria Sophia Antipolis that is working on methods and tools to manage concurrency and time at different levels of abstraction.

The rest of the document is designed as: the overview of the recent work in contact tracing is explained in Chapter 2, Chapter 3 present the detail introduction of the protocol. Messages and communication of the entities are explained in Chapter 4. Chapter 5 shows the implementation of the entities and the future work. Conclusion and discussion are explained in Chapter 6.

Chapter 2

Literature Review

Since last year after the rise of the SARS-CoV-2 virus there are several research work done in detecting the virus. As the most common way of the detecting and tracing virus is done through Digital Contact Tracing. This section provides a quick overview of some of the research done in Digital Contact Tracing.

Contact Tracing has been actively used in Europe since the 16th century to contain epidemic disease. The principles remain the same today whether carried out by phone, mail or personal contact. The aim is to identify the origin of the infection and to where, or to whom, it has been transferred. If receipt of this information is followed immediately by isolation, treatment and aggressive decontamination, it can lead to containment and the gradual elimination of the disease itself.

Digital Contact Tracing is used to detect and trace the human connections to an infected person and informing the person to go into isolation. The Digital Contact Tracing tools/protocols uses GPS and Bluetooth feature of the smartphone. The idea was initially discussed back in 2007, where in 2014 the idea was proven for the first time using Bluetooth. This idea came under the spotlight after the COVID-19 pandemic. There are some work done in Digital Contact Tracing for COVID-19 pandemic. Currently used Digital Contact Tracing systems are *synchronous systems*. This type of the tracing need the two parties at the same place at the same time. Synchronous contact tracing are mainly of two types:

- **Centralized:** In this method the contact details and history are stored at the National Health Authority or at main server.
- **Decentralized:** In this method the contact details and history are stored and managed by the clients in the network.

We will provide a quick overview of the Digital Contact Tracing systems below.

2.1 ROBERT

Robust and Privacy-preserving Proximity Tracing (ROBERT) [3] protocol is a secure and robust scheme for Digital Contact Tracing. COVID-19 is hard to trace as many people can be the carriers. In this case a smartphone can be helpful, with the help of short range communication, i.e Bluetooth in smartphone an application was built to trace the potential virus carriers. ROBERT covers all the aspects of the security of the information shared and the user privacy of the user data. This mechanism needs the two parties to be in range of the Bluetooth communication. This scheme notify the user about their close contact in N days to some potential virus infected people. When the user voluntarily install the application in their smartphone, the application register the user with the National Central Authority. The Authority set some random name to for the user and share it to the user application. They referred this name as 'false name'. These names are generated using cryptographic function which involve the secret key associated with the application. The protocol make it sure that the user name, location and other useful information is not shared.

2.2 Pan-European Privacy-Preserving Proximity Tracing (PEPP/PT)

Pan-European Privacy-Preserving Proximity Tracing (PEPP/PT) [4] is a protocol used for Digital Contact Tracing for COVID-19 pandemic. The goal of the protocol is to provide a common basis for management systems that can be integrated into national public health responses to the COVID-19 pandemic. The PEPP-PT approach is being created by a multi-national European team. It is a privacy-preserving digital proximity-tracing approach, which is in full compliance with GDPR and can also be used when traveling between countries. Unlike DP-3T, described below, it is based on a centralized approach for tracing. So the main server modify the contact logs and inform single user about their contact in near day with tested positive patient. This approach compromises the privacy of the user. On the back-end server processes pseudonymous personal data that would eventually be capable of being re-identified. The positive point is that it

provide human in the loop checks and is verified from National Health Authority.

2.3 Decentralized privacy-preserving (DP-3T)

Decentralized privacy-preserving (DP-3T) [5] is a Digital Contact Tracing system which trace the close physical contacts of the people with virus infected people, without revealing the identity and location information of the user. DP-3T is highly secure and preserve the privacy of the user according to GDPR rules. The DP-3T require the user to have a smart application installed on the phone which will broadcast the pseudo-random ID and also receive the IDs of all other user in close proximity. When a user is tested positive she will upload the information with her ID to the main server and all the user who were in contact or close proximity with her they will check the information corresponding to the IDs their phone recorded. All this will happen inside smart application installed on smartphone. This system is highly secure and provides protection against malicious use of user data and tracking of the user (location and user credentials). The communication in this system is done using Bluetooth technology.

2.4 Exposure Notification System by Google/Apple

Exposure Notification System (ENS) [6] is another Digital Contact Tracing tool developed by Google and Apple. This tool trace and scan the connection using Bluetooth. ENS is using DP-3T protocol. The goal was to make a tool which will be used widely for tracing independent of the National Health Authorities. The users decide whether they wish to opt-in to the Exposure Notifications and the system does not collect any location information from the device. This application is installed on the user phone without the consent of the user. Although it remain inactive until the user install some other third party application that works in collaboration with this application. Table 2.1 summarizes the available approaches and their applications.

All theses currently used tools and protocols performs a synchronous tracing, which needs two parties to be *at the same place* and *at the same that moment* and with in the Bluetooth communication range. There is need of a protocol which can remove the constraint of the time and space. Asynchronous Contact Tracing (ACT) [7] is an ETSI standard protocol which trace virus not humans i.e trace the things/materials hosting virus not humans. So for ACT there is no constraint of time and space. The SARS-

Table 2.1: Table of Available Approaches of Digital Contact Tracing

Summary of Existing Approaches of Digital Contact Tracing				
Name	**Approach**	**Applications**	**Technology Used**	**OS-Supported**
ROBUST	Centralized	StopCOVID	Bluetooth Low Energy	Android/IOS
PEPP/PT	Centralized	PEPP-PT	Bluetooth Low Energy	Android/IOS
DP3T	Decentralized	ENS Google and Apple	Bluetooth Low Energy	Android/IOS
ENS	Decentralized	Stop Corona, Hoia, Corona-Warn-App, COCOA	Bluetooth Low Energy	Android/IOS

CoV-2 virus is known to stay viable for a variable time on a hard surface, depending from the nature of the surface, the virus concentration, the temperature, the humidity conditions, the exposition to sun light. This time can vary from few hours to several days, some examples are 2-3 hours on paper, 4 hour on copper, 3-4 days on plastic and steel, 7 days on face masks, and even more in specific climate conditions [8] [9] [10] [11]. ACT perform test on the material infected with virus regardless of who infected it. As the user information is not involved in this process which makes this protocol highly privacy preserved.

Chapter 3

Asynchronous Contact Tracing (ACT)

In this chapter we will explain the asynchronous contact tracing protocol in detail. The main component involved in the ACT will be explained.

3.1 Introduction of ACT

Asynchronous Contact Tracing (ACT) trace IoT connected objects that "host" the SARS-CoV-2 virus instead of the people that got infected by the SARS-CoV-2 virus. Unlike Synchronous Contact Tracing, ACT does not require the two person to be at the same location and at the same time. ACT remove the limit of distance and time. ACT traces the virus instead of the people who carry the virus. ACT traces and test the surfaces, objects or locations that have been contaminated by virus. The SARS-Cov-2 virus can stay contagious for some time on the objects depending on the material of the object, virus, temperature humidity conditions and exposition to sunlight. This is said that virus can stay on the paper for 2-3 hours, 4 hours on copper, 3-4 days on plastic and steel and 7 days on face masks [8] [9] [10] [11].

ACT can be deployed in any environment where the humans share same physical space like supermarkets, train stations, fitness centres, offices, schools etc. Where the test for the detection of the virus will be done on different objects depending on the type of department. ACT will be used as a tool to recommend a *selective lockdown* instead of a *full lockdown*, which have huge impact on the personal life of human as well as the economic situation.

ACT uses the Prof. Dorfman, so called, *Group Testing* [12] methodology, as we are interested in the absence or presence of the virus. The reason behind this is that humans

are not able to know about the virus infected objects or persons. Dorfman group testing was testing humans in a group. For example some blood is taken from ten people and then mixed together and tested. If the result is negative all the 10 humans are declared negative and save 9 testing kits. But if the test result is positive, then all the samples taken will be tested separately. The same approach is followed in ACT, different objects samples are tested and with modern digital IoT technology this may provide a new and effective forecast for selective lockdown. ACT will make sure the result of the testing is communicated to the people of the geographical area which can be declare safe or unsafe based on the test result by Public Health Authority.

3.2 ACT Process

ACT is a Digital Contact Tracing protocol which work as follows:

- A number of common wireless access points which are referred as ACT Peripheral Services, are installed in the location. For example, in corridors of the supermarket, classrooms in schools, train stations, etc. These devices will transmit its unique identifier called MAC Identifier BSSID and unique human readable name known as WiFi identifier SSID.
  - Basic Service Set Identifiers (BSSID) is a unique identifier of the network device. This is unique for every device thus the replication of the BSSID is hard which makes the *spoofing attack* harder.
  - Service Set Identifiers (SSID) is a human readable name of the network. In ACT, this name should start with "ACT-" followed by any string.
- A smartphone will record the BSSID and SSID of the ACT Peripheral Service. The user will then use this BSSID to ask for the forecast from the ACT Control Service. The geographical location visited by the user will not be shared with any third party to comply with GDPR rules [13].
- For collecting the test samples and results from the laboratory, ACT introduce a special service, called Detection Service. It is referred as ACT Detection Service in the ACT protocol. The test will be done according to the material of the object. These test results will be then send to the local or Public Health Authority (PHA).

- The results from the ACT Detection Services are sent to the ACT Local Service and then ACT Control Service. The results are associated with the BSSID and the localization of the ACT Peripheral Service. The information received are interpreted by the Public Health Authority into a virus forecast for the location. The forecast is communicated to the users using mobile or web applications.

- Smartphone application or Web-based application are used to transmit the forecast from the PHA. The results are associated with the location and then with the BSSID recorded by the user phone.

- In case the owner of the smartphone is tested positive, the user will send the test result along with the date of the test and the BSSIDs visited to the ACT Control Service. An extra layer of authentication can be done at this level to avoid the false declaration of the test result.

3.3 ACT Use Case

Let imagine Alice is infected with virus but is not aware of it, went to supermarket to do groceries around 10am. Meanwhile, she touches different objects and went through different corridors and sneezes or coughs. So the corridors she went through or the objects she touches are infected with SARS-CoV-2 virus. Now if Bob comes to the same supermarket later the day let say 11:30am. Bob is not infected with virus. He unfortunately touches the same items Alice did and went through the same corridors and get infected with virus.

In the above scenario Synchronous Digital Contact Tracing Protocols cannot work. As both the parties were not in the same place at the same time so there is no exchange of pseudonyms. Also if Alice went back and did a test and the result is positive there is no way to inform Bob about it who is infected with the virus also.

Asynchronous Contact Tracing protocol helps in this case in such a way. Alice and Bob phones will record the BSSIDs of the corridors they went through in the supermarket. Later the day when those objects are tested and the found infected. The PHA will inform the supermarket as well as alert the user in the localization. Also Alice and Bob will query ACT Control Service with the location and the BSSIDs their phone recorded, will get to know that they have been to the infected place. PHA will give a recommendation message along the response.

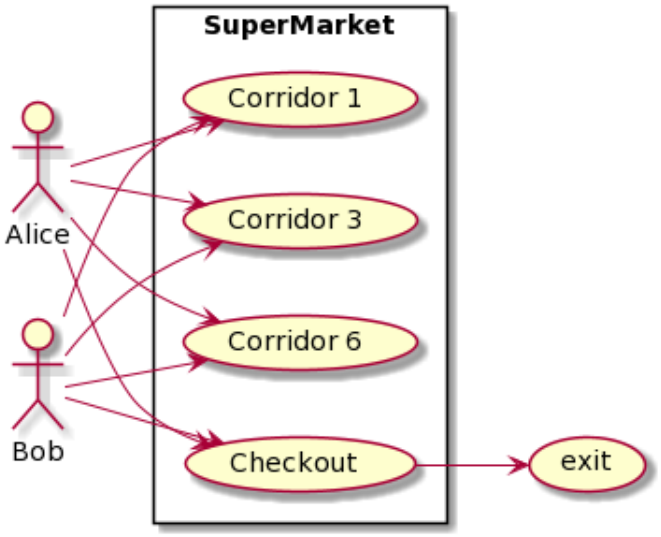

Figure 3.1: ACT Use Case

3.4 ACT Protocol Architecture and Functionalities

The ACT protocol comprises of six main elements. ACT is deployed in ETSI oneM2M standard framework [2]. The ACT architecture is depicted in Figure 3.2. In this section we will explain in details all the entities of ACT.

3.4.1 ACT Peripheral Service

The ACT Peripheral Service, placed in a specific location, such as a supermarket corridor, a public toilet, a metro station, a fitness or hotel room etc. has the capability to transmit, using WiFi technology, the following information such as its unique WiFi identifier (also known as the MAC identifier BSSID) and its unique WiFi human-readable network name (also known as the WiFi identifier SSID) [1]. The scope of this entity is to provide frequently with a configurable periodicity the BSSID identifier that permits the human to record the passage in a determined correlated area (e.g. a corridor, a metro station or a metro car, a restroom, an open garden, an office, etc.).

It shall broadcast its BSSID in order to allow the human handheld devices to record it, and by pattern matching, to discover the risk of having been infected by staying in its proximity area. It shall also broadcast its SSID (always starting with the “ACT-” prefix) in order to allow the ACT Smart Mobile Application to detect which related BSSID identifier should be memorized in the internal memory of the smartphone.

It shall be connected to one or more ACT Detection Service to correlate the de-

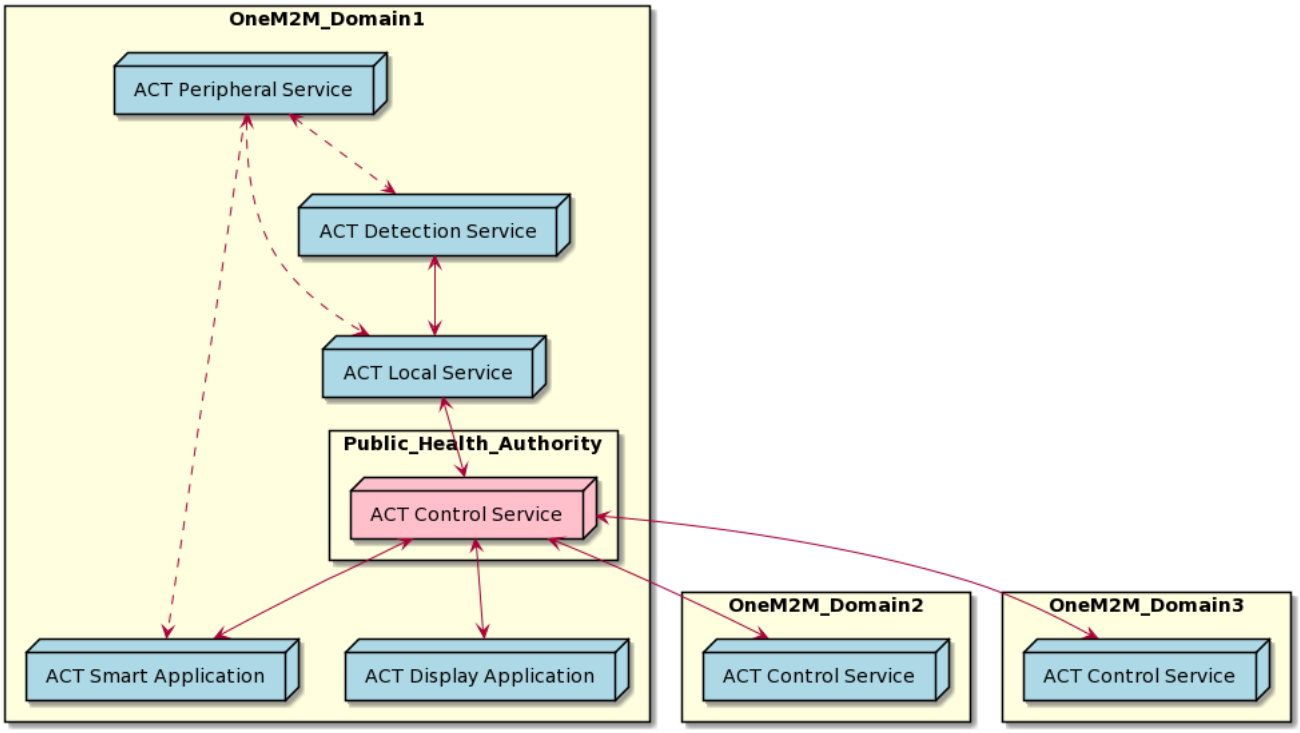

Figure 3.2: ACT Protocol Architecture

tection of the contamination with the proximity area where ACT Peripheral Service is deployed.

3.4.2 ACT Detection Service

The ACT Detection Service informs the one ACT Local Services about the detection of the contamination in one of the proximity areas associated to one or many ACT Peripheral Service [1].

This service is connected to one or many ACT Peripheral Service and one ACT Local Service and inform ACT Local Service about the detection of the contamination in one of the proximity areas associated to one of the ACT Peripheral Service.

The biological process and the kind of test used to check the material for contamination (e.g. HEPA air filter, waste-water filter, dirty cleaning tools and water, etc.) are non-pertinent to the ACT Technical Specification: nevertheless the exchange of specific configuration and monitoring information related to the detection tools are supported.

3.4.3 ACT Local Service

The ACT Local Service receive information from one or many ACT Detection Services and forward the information to the ACT Control Service [1].

The scope of the entity is to receive the messages from ACT Detection Node related to it and forward it to the ACT Control Service. It also receive the necessary indications in order to behave according to the Public Health Authority policies for the ACT Peripheral Service configuration of BSSID and SSID identifiers.

It shall exchange the BSSIDs of all the ACT Peripheral Services involved in the detection to the ACT Control Service. Also it should exchange configuration information with ACT Control Service.

3.4.4 ACT Control Service

An ACT Control Service will receive the information from all the ACT Local Services related to ACT Detection Services located in the area of pertinence of the ACT Control Service, and will provide to it information, according to the Public Health Authorities policies. It will exchange its monitoring and configuration information with the one or many ACT Local Services and with the many ACT Smart Application and ACT Display Application. It will coordinate and communicate the information about the areas of detection of the virus in other Municipalities, Departments, Regions, or Nations via others ACT Control Services. This enables ACT to be a genuine, fully-fledged forecast tool [1].

The scope of this entity is mainly receive the information from the ACT Local Services related to contamination according to the Public Health Authority policies. The ACT Control Service also represents the point of interaction with the ACT users.

It shall provide, upon request, the identifier of the one or several ACT Peripheral Services announcing contamination to the ACT Smart Mobile Application with additional information about indications and suggestions according to the Public Health Authority policies (e.g. the suggestion of avoiding certain areas or to perform a human test verification if certain conditions of exposure are met according to the detection of the virus in the areas covered by one or many ACT Peripheral Services).

3.4.5 ACT Smart Application

An ACT Smart Mobile Application is the digital tool available to the users to monitor the level of contamination. It collects the identifier from the one or many ACT Peripheral Services and periodically compares it with the identifier-related published by the ACT Control Service, with an associated time and with a forecast, synthesized with the help of the Public Health Authority, about potential contaminated areas [1].

The scope of this entity is mainly receive the information from the ACT Control Service, collect the identifiers from the ACT Peripheral Services and periodically compares this information with the ones from the ACT Control Services about the contaminated areas.

3.4.6 ACT Display Application

An ACT Display Application is hosted by an internet-connected device (Tablet, Personal Computer, Smart TV, etc.) that communicate with the ACT Control Service to query and receive feedback about the status of the situation in specific geographical areas. By delegation, it can also query other (international) ACT Control Services [1].

It shall support the user about the discovery of its own risk according to area of interest, e.g. location visited in the past or areas planned to be visited in the future. It does not collect any BSSID information about the visited Access Points.

Chapter 4

Communication in ACT

In this chapter we will discuss about the communication between the ACT entities. Also we will define different messages for each entities. For each entity we will define a set of message they will receive and send.

4.1 ACT Messages

4.1.1 Information Broadcasted by ACT Peripheral Service

The ACT Peripheral Service will broadcast its BSSID and SSID. The ACT Smart Mobile Application will listen to the broadcast message. As we have seen in Chapter 3, that ACT Peripheral Service is a WiFi access point so the communication medium is wireless [15]. The BSSID is a string and will be referred as ACT Peripheral-Service-ID. The SSID is configured by the ACT Local Service with a prefix "ACT-". The reason behind using the prefix of SSID as "ACT-" is that the ACT Smart Mobile Application will filter SSID and remember the BSSID for the SSID which starts with "ACT-". The ACT Smart Mobile Application in the user phone should store the following information:

- BSSID/SSID, as explained before.
- TIMEIN: the time when the BSSID/SSID is detected by the smartphone.
- TIMEOUT: the time the smartphone lost signal to the ACT Peripheral Service.

Table 4.1 specify the information broadcasted by ACT Peripheral Service.

Status	Active
Test-Time	2021-07-08T11:28:27.176750
Test-Result	3

Figure 4.1: Example of message send from ACT Detection Service to ACT Local Service

4.1.2 Information Exchanged Between ACT Detection Service and ACT Local Service

This communication will be two way communication where ACT Detection Service will send the information about the tests and results of the tests performed on the objects. While ACT Local Service will in return send some messages for initiating the test or restart the detection process and a time interval between two tests results. Table 4.2 specify the messages send from ACT Detection Node to ACT Local Service and Figure 4.1 shows the example message send from ACT Detection Service to ACT Local Service. Table 4.3 shows the Information sent from ACT Local Service to ACT Detection Service and Figure 4.2 shows an example message send from ACT Local Service to ACT Detection Service.

Command	Restart
Test-Interval	3

Figure 4.2: Example message send from ACT Local Service to ACT Detection Service

4.1.3 Information Exchanged Between ACT Local Service and ACT Control Service

The communication between ACT Local Service and ACT Control Service is an *n to 1* communication. There can be many ACT Local Services connected to one ACT Control Service. ACT Local Service collect the data from one or more ACT Detection Services connected to it and send this data with geographical information to the ACT Control Service. Where the PHA interpret the data and take the necessary measures. The location information is shared using Geo-hash Code System [16]. Geo-hash is one of the unique code used in many application. It takes the latitude and longitude and return a random string. This code is precise and there is precision parameter to set the length of the string. The longer the string the more precise is the location. Table

4.4 will show the detail of the messages send from ACT Local Service to ACT Control Service and Figure 4.3 shows an example message.

Peripheral-Service-Id	C8:60:00:4C:27:A5
Location	ezs42e44yx96
Location-Service-Info	https://NationalHealth.com/forecast
Test-Result	3
Test-Time	2021-07-08T11:28:27.176750
Disinfection	☑ true
Disinfection-Time	2021-07-08T11:28:27.176750

Figure 4.3: Example message from ACT Local Service to ACT Control Service

4.1.4 Information Exchanged Between ACT Smart Mobile Application and ACT Control Service

The ACT Smart Mobile Application send a query to ACT Control Service with the location parameter as shown in Table 4.5. The ACT Control Service send a detail response to the user with the fields mentioned in Table 4.6. The example of the response message is shown in Figure 4.4.

The ACT Smart Mobile Application shall periodically send a query the ACT Control Service. The periodicity of the request is configured in the application, and can be modified by the ACT Control Service by means of application software version upgrades.

The precise geographical position of the smartphone shall not be transmitted to the ACT Control Service. In case of absence of underlay network connectivity, the ACT Smart Mobile Application will keep track of all visited geographical areas, and, as soon as the network connectivity will be available, the ACT Smart Mobile Application will issue a sequence of queries related with such areas.

The CONTROL-SERVICE-IDs are configured in the ACT Smart Mobile Application: many CONTROL-SERVICE-IDs can be then stored in the ACT Smart Mobile Application; this allow to correlate the ACT Control Service with the given Municipality/Department/Region/Country it is currently depending in.

The ACT Control Service will send back a detailed response. The response also includes the time periods of RED-FORECASTS associated to each of the reported PERIPHERAL-SERVICE-ID. The relevance of the contamination is assessed by the

ACT Control Services according to Public Health Authority policies; only RED cases are reported.

Additionally, if an user is tested positive to the virus, it may add the information to the ACT Smart Mobile Application, to inform the ACT Control Service of the potential contamination of visited locations, by sending the information described in Table 4.9 and Figure 4.5 shows the example of message.

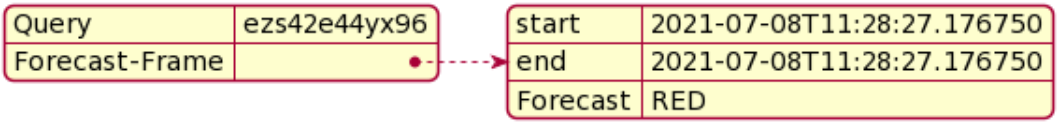

Figure 4.4: Example query from ACT Display Application to ACT Control Service

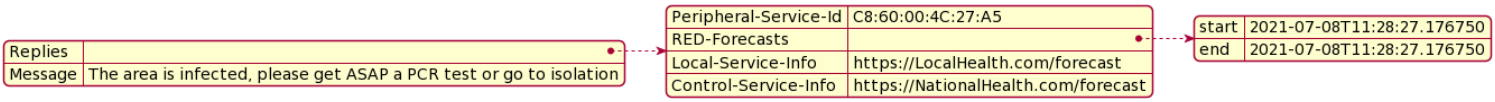

Figure 4.5: Example response from ACT Control Service to ACT Smart Mobile Application

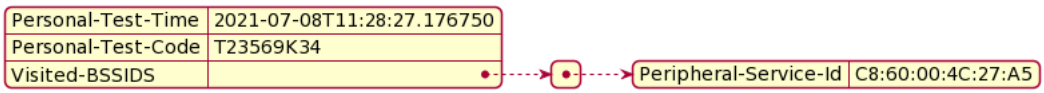

Figure 4.6: Example message from ACT Smart Mobile Application to ACT Control Service

4.1.5 Information Exchanged Between ACT Display Application and ACT Control Service

The web-based ACT Display Application (Personal Computer, Tablet, Smart TV, etc.) sends queries to the ACT Control Service for providing information. The message details are shown in Table 4.10 and Figure 4.6.

The ACT Control Service will send a detailed response to the ACT Display Application Query. The response fields are described in Table 4.11 and example of response is shown in Figure 4.7.

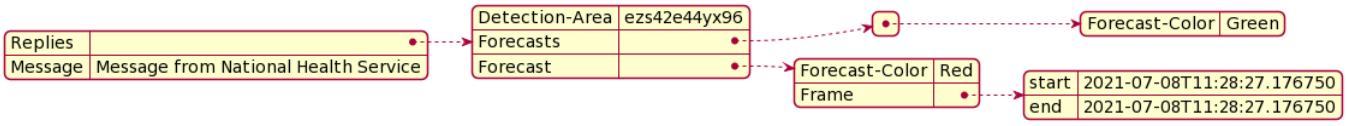

Figure 4.7: Example response from ACT Control Service to ACT Display Application

4.1.6 Information Exchanged Between Different ACT Control Services

The ACT Control Services should be able to communicate each other's to share the ACT related information across the domains of the different Public Health Authorities. The target ACT Control Service(s) is(are) selected according to the geographical area of competence, and the query is then sent on the corresponding communication framework interface. One ACT Control Service will send a query with location field to another ACT Control Service as shown in Table 4.14.

The ACT Control Service shall respond with a list of replies specified in the following Table 4.15, where each reply is defined in Table 4.16. The relevance of the contamination is assessed by the relevant ACT Control Services according to relevant Public Health Authorities policies. The example of the response message is shown in Figure 4.8.

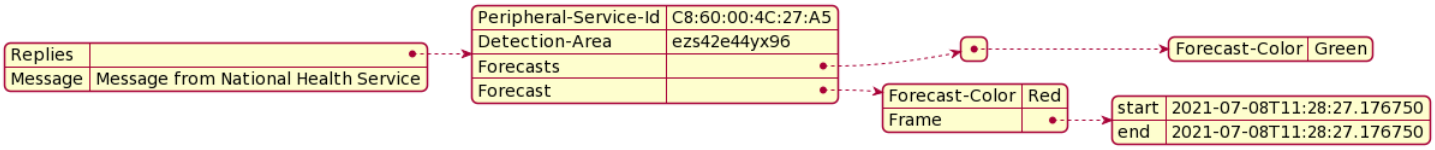

Figure 4.8: Example response from ACT Control Service to ACT Control Service

Table 4.1: Information sent from ACT Detection Service to ACT Local Service

Parameter Name	**Type**	**Description**
STATUS	STRING: ACTIVE; SLEEPING; OUT-OF-SERVICE; RESTARTING; MAINTENANCE-REQUESTED; FAULT	This parameter shall be mapped according to the following: ACTIVE: the Detection Service is operative; SLEEPING: the Detection Service has the energy saving status activated (e.g. when the associated shop is currently closed); OUT-OF-SERVICE: the Detection Service is still connected but it cannot become operative (e.g. it is subject to a maintenance procedure); RESTARTING: the Detection Service is in the process of becoming operative; MAINTENANCE-REQUESTED: the Detection Service require a maintenance (e.g. for the refilling of the test reagents, for the calibration, or for other tuning reasons); FAULT: the Detection Service has found itself to be faulty.
TEST-TIME	ISO 8601 [14]	The time of the collection of the test sample shall be reported in the TEST-TIME parameter. This parameter is absent if the Detection Service answers to a STATUS command and there is no TEST-RESULT to be reported.
TEST-RESULT	NATURAL	The time of the collection of the test sample shall be reported in the TEST-TIME parameter. This parameter is absent if the Detection Service answers to a STATUS command and there is no TEST-RESULT to be reported.

Table 4.2: Information Broadcasted by ACT Peripheral Service

Parameter Name	**Type**	**Description**
PERIPHERAL-SERVICE-ID	STRING	BSSID of the ACT Peripheral Service. For any other communication medium it may vary.

Table 4.3: Information sent from ACT Local Service to ACT Detection Service

Parameter Name	**Type**	**Description**
COMMAND	STRING: RESTART; SHUTDOWN; SLEEP; STATUS-REQUEST; TEST-START; TEST-STOP;	This parameter shall be mapped according to the following: RESTART: the Detection Service shall restart; SHUTDOWN: the Detection Service shall shutdown; SLEEP: the Detection Service shall activate the energy saving status; STATUS-REQUEST: The detection service shall respond providing the STATUS Parameter; TEST-START: the Detection Service shall initiate performing the tests, according to the given TEST-INTERVAL indication; TEST-STOP: the Detection Service shall stop performing the tests.
TEST-INTERVAL	NATURAL	This parameter shall be mapped according to the following:0: the test shall be executed continuously; 1 to N: time interval, expressed in seconds, from the time of the last test if available; in case the time from the last test is not available, the test shall be executed immediately.

Table 4.4: Information Exchanged between ACT Local Service and ACT Control Service

Parameter Name	**Type**	**Description**
PERIPHERAL-SERVICE-ID	STRING	BSSID of the ACT Peripheral Service. For any other communication medium it may vary.
LOCATION	STRING	It shall contain the geographical area, represented using the Geo-hash Code System [16] of the place where the sample reported in the provided TEST-RESULT has been taken.
LOCAL-SERVICE-INFO	URL	URL pointing to customized information provided by the Local Service (valid for that specific Peripheral Service).
TEST-RESULT	defined in Table 4.2.	defined in Table 4.2.
TEST-TIME	defined in Table 4.2.	defined in Table 4.2.
DISINFECTION	BOOLEAN	TRUE Indicates that: active virus is not expected to be present, and traces of not active virus have been reasonably removed.
DISINFECTION-TIME	Time as defined in ISO 8601 [14].	Time of completion of the disinfection.

Table 4.5: Query from ACT Smart Mobile Application to ACT Control Service

Parameter Name	**Type**	**Description**
QUERY	STRING	It shall contain the geographical area represented using the Geo-hash code System [16]. It includes the area visited by the device hosting the Smart Mobile Application.

Table 4.6: Response by ACT Control Service to ACT Smart Mobile Application

Parameter Name	**Type**	**Description**
REPLIES	LIST Of REPLY	List of replies indicating the Peripheral Services with the related RED forecasts. Defined in Table 4.7.
MESSAGE	STRING	When present, it shall be displayed by the Smart Mobile Application to provide guidance to the user.

Table 4.7: REPLY

Field Name	**Type**	**Description**
PERIPHERAL-SERVICE-ID	Defined in Table 4.1.	Defined in Table 4.1.
RED-FORECASTS	LIST	Defined in Table 4.8.
LOCAL-SERVICE-INFO	URL	URL pointing to customized information provided by the Local Service (valid for that specific Peripheral Service).
CONTROL-SERVICE-INFO	URL	URL pointing to customized information provided by the Control Service (valid for that specific Peripheral Service).

Table 4.8: RED-FORECASTS

Field Name	**Type**	**Description**
FRAME	(Time, Time) as in ISO 8601 [14].	This parameter indicates the RED time frame of the corresponding Peripheral Service.

Table 4.9: Message from ACT Smart Mobile Application to ACT Control Service

Parameter Name	**Type**	**Description**
PERSONAL-TEST-TIME	Time as defined in [14].	The time when user is tested positive for virus.
PERSONAL-TEST-CODE	STRING	Reference number of the test ensuring that test is organized by PHA.
VISITED-BSSIDS	LIST	All the Peripheral Service ids user visited.

Table 4.10: Query from ACT Display Application to ACT Control Service

Parameter Name	**Type**	**Description**
QUERY	STRING	Geographical area of interest represented using the Geo-hash code System [16].
FORECAST-FRAME	((Time, Time), STRING)	It shall contain two Times, i.e. the time-interval for which a forecast is queried and a STRING that can be: RED in case the query is asking only for notifying PERIPHERAL-ID with RED forecast; ALL in case the query is asking for notifying PERIPHERAL-ID with all the kind of forecast.
VISITED-BSSIDS	LIST	All the Peripheral Service ids user visited.

Table 4.11: Response from ACT Control Service to ACT Display Application

Parameter Name	**Type**	**Description**
REPLIES	LIST OF REPLIES	defined in Table 4.12
MESSAGE	STRING	When present, it shall be displayed by the Display Application to provide guidance to the users.

Table 4.12: REPLIES

Field Name	**Type**	**Description**
DETECTION-AREA	STRING	Geographical area of application of associated forecast, represented using the Geo-hash code System [16].
FORECASTS	LIST OF FORECASTS	List of FORECAST associated to the DETECTION-AREA defined in Table 4.13.

Table 4.13: FORECASTS

Field Name	**Type**	**Description**
FORECAST-COLOR	STRING	The meaning of each color code is defined by the Public Health Authority. Possible Colors are : RED GREEN YELLOW GREY
FRAME	(Time, Time) as in [14].	This parameter indicates the time frame of the FORECAST.

Table 4.14: Query from ACT Control Service to ACT Control Service

Parameter Name	**Type**	**Description**
QUERY	STRING	It shall contain the geographical area represented using the Geo-hash code System [16]. It includes the area visited by the device hosting the Smart Mobile Application.

Table 4.15: Response from ACT Control Service to ACT Control Service

Parameter Name	**Type**	**Description**
REPLIES	LIST OF REPLIES	Defined in Table 4.15.
MESSAGE	STRING	It provide guidance between Control Services.

Table 4.16: REPLIES

Field Name	**Type**	**Description**
PERIPHERAL-SERVICE-ID	Defined in Table 4.1.	Defined in Table 4.1.
DETECTION-AREA	STRING	Geographical area of application of associated forecast, represented using the Geo-hash code System [16].
FORECASTS	LIST OF FORECASTS	List of FORECAST associated to the DETECTION-AREA defined in Table 4.13.

Table 4.17: FORECASTS

Field Name	**Type**	**Description**
FORECAST-COLOR	STRING	The meaning of each color code is defined by the Public Health Authority. Possible Colors are: RED GREEN YELLOW GREY.
FRAME	(Time, Time) as in [14].	This parameter indicates the time frame of the FORECAST.

Chapter 5

ACT Implementation

In this chapter, we will show the implementation process of the ACT protocol and its entities. Also we will see how we can deploy the protocol in real ETSI oneM2M [2] environment.

5.1 ACT Entities and oneM2M Resources

In order to implement the ACT services,, we make use of oneM2M specifications to support the communication and interopertability. To make use of oneM2M functional architecture [17] we will use the following ETSI oneM2M concepts:

- Application Entity (AE): An AE is an application layer entity that implement the M2M application service logic. Each instance of a software is an AE [17].
- Common Service Entity (CSE): A CSE works as an instance that manage and provide the common services in the network [17].
- Application Dedicated Node (ADN): An ADN is a node that contain at least one AE and does not contain any CSEs [17].
- Application Service Node (ASN): An ASN is a node that contain at least one CSE and one AE inside [17].
- Infrastructure Node CSE (IN CSE): An IN CSE contains at least one CSE and zero or more AEs [17].
- Mca: This reference point is used for communication between AE and CSE [17].
- Mcc: This reference point is used for communication between two CSEs [17].

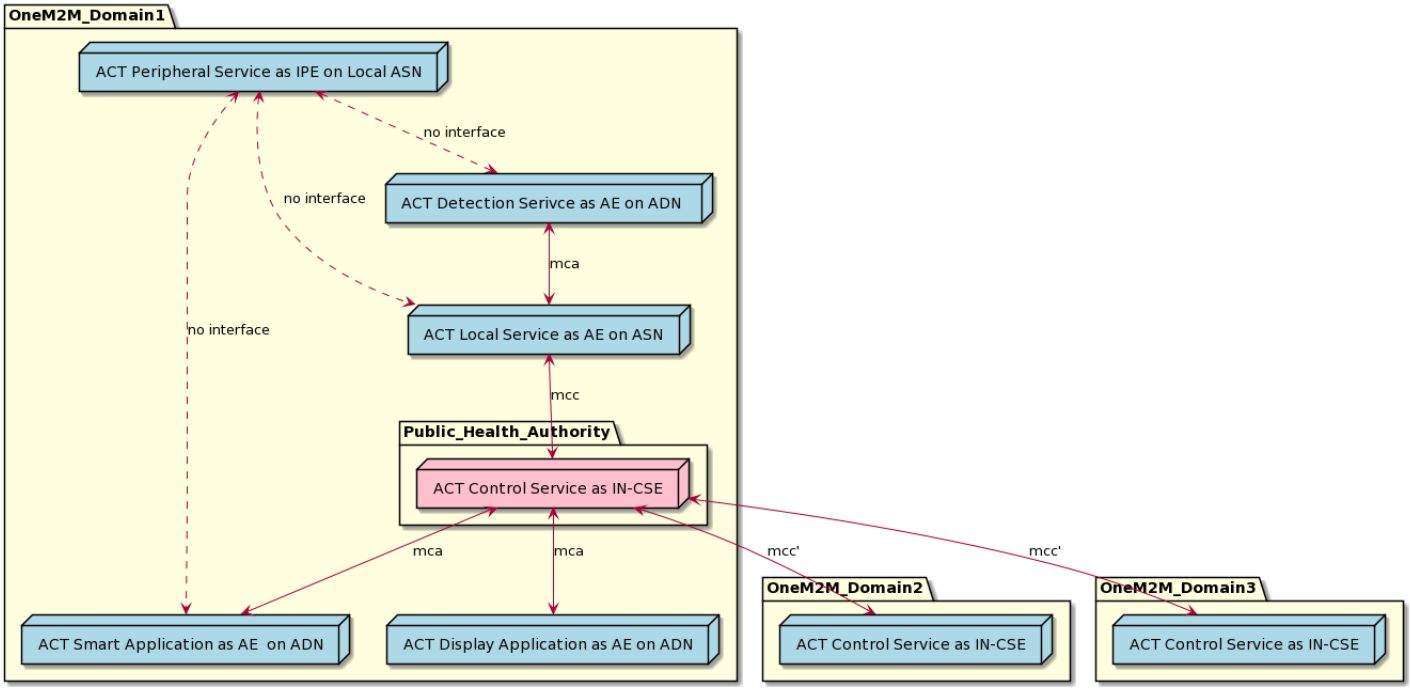

Figure 5.1: Mapping of ACT Entities to oneM2M Elements

- Mcc': This reference point is used for communication between CSEs inside IN CSE [17].

The Figure 5.1 abovesshows the mapping of ACT entities into common oneM2M framework. For ACT Peripheral Service the interface depends on the configuration of device.

5.2 Implementation Setup

For this implementation and experiment we are using Dell Core i7 vPro 7th Generation with following specifications:

- 32GB of RAM
- Windows 10
- 64 bit Operating System

For connection to ICON TIM server we were not able to connect through Inria WiFi connection, we requested for a DMZ port to get access to the ICON TIM. Thanks to the Inria IT team for their timely help.

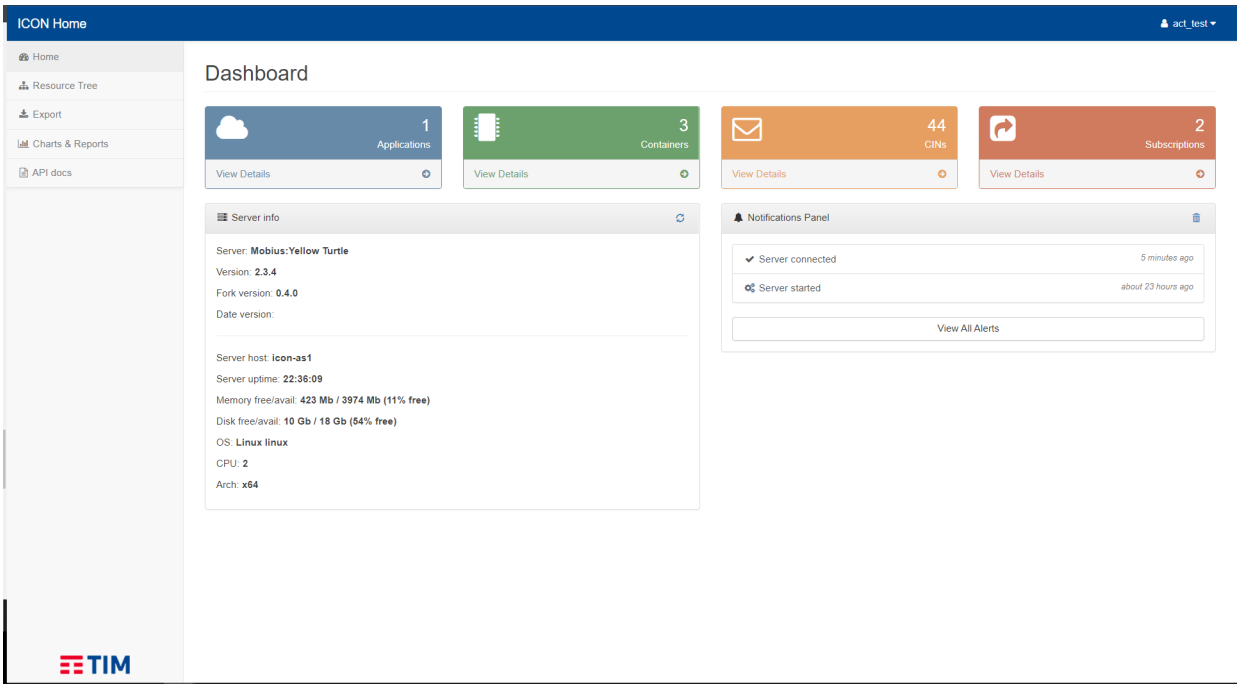

Figure 5.2: ICON TIM GUI

5.3 Software Used in Implementation of ACT

We experimented the ACT protocol in ETSI oneM2M framework by using a platform connected directly to one oneM2M cloud. Telecom Italia Mobile (TIM) provides an ETSI oneM2M based middleware implementation platform *ICON*. Alongside ICON, we used Python programming language to interact with the ICON. Below sections provide complete details of the implementation.

5.3.1 ICON Platform by Telecom Italia Mobile (TIM)

ICON API is an implementation of ETSI oneM2M infrastructure. ICON allow user to deploy their protocol/network on ICON oneM2M cloud. User is issued a username and password by admin. Each tenant can have one or more Application Entity associated. ICON use REST structure to communicate with user or other servers/websites. The GUI of ICON is shown in the Figure 5.2.

In ICON we can create containers, container instances, subscription and access control policies for containers. Containers (CNT) are created under AE, which have specific content type and content. Container whenever receive some new information it creates a Container Instance (CIN) and store the value in it. In ICON one can also

define an access control policy per every container. Subscription (SUB) are used to subscribe on resource, whenever there is modification under the subscribed container a notification is send to the end node about the change. Also ICON allow to group containers of same resource together using Group (GRP). Section 5.3 will explain about how to create resource tree and interact with ICON.

5.3.2 Python Programming Language

With ICON we used Python programming language to interact with ICON from outside. We used different python libraries for this implementation. Requests library [18] in Python allow us to send and receive HTTP request to ICON. *Pygeohash* Python module [19] is used for converting the location parameter into an hash code.

5.4 ACT Entities Implementation

5.4.1 ACT Detection Node

For ACT Detection Node on ICON we created a container by sending a POST request to the ICON using Python as shown below:

```
import requests
import json
from    requests.auth import HTTPBasicAuth

API_endpoint= "https://icon-lab.tim.it/oneM2M/act_test"
data = """{
    "m2m:cnt": {
    "rn": "ACT_DetectionNode",
    "lbl" : ["Detection", "Data"]}
}"""
headers= {
   'X-M2M-Origin': 'act_test_prod',
   'X-M2M-RI': str(int(time.time())),
   'Accept': 'application/json',
   'content-Type': 'application/vnd.oneM2M-res+json;ty=3'
 }
```

```
r = requests.post(url= API_endpoint,
    auth = HTTPBasicAuth('username', 'password'),
    headers = headers,
    data = data)
r.json()
```

In the above code we send a POST HTTP request to ICON. In the data "rn" stands for resource name , and "lbl" is label. Now to write the data to the detection node we will create a container instance under the container ACT_DetectionNode.

```
import requests
import json
from requests.auth import HTTPBasicAuth
import time

detection_service_id = 'Skf1nYQGcd'
status = {'active',
  'sleeping',
  'out-of-service',
  'restarting',
  'maintenance-requested',
  'fault'}
test_time = datetime.now().isoformat()
# get the value of the test result
test_result = input("enter an integer value")

data = {peripheral_id': '4A:54:4C:23:67:B6',
'status': 'active',
'test_time': test_time,
'test_result': test_result}
new_dict = {'m2m:cin': {'cnf': 'application/json:0','con': data}}
payload = json.dumps(new_dict)

# create a connection to the server
```

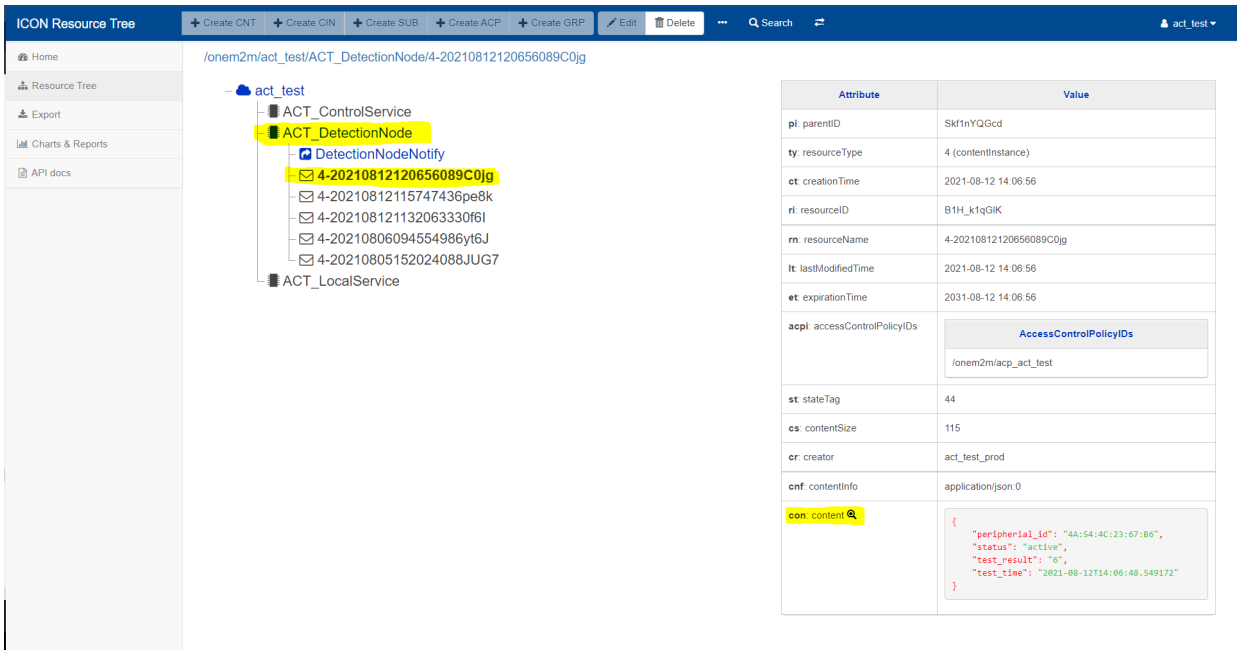

Figure 5.3: ACT Detection Service in ICON

```
API_endpoint = "https://icon-lab.tim.it/oneM2M/act_test/ACT_DetectionNode"
headers = {'X-M2M-Origin': 'act_test_prod',
  'X-M2M-RI': str(int(time.time())),
  'Accept': 'application/json',
  'content-Type': 'application/vnd.oneM2M-res+json;ty=4'}
r = requests.post(url = API_endpoint,
  auth = HTTPBasicAuth('username', 'password'),
  headers = headers, data = payload)
r.json()
```

The results of the above code can be seen on ICON GUI in Figure 5.3. In the Figure 5.3 we can see that the ACT_DetectionNode container is created under Application Entity (AE) and a container instance is created under the container, content of the container represented in JSON format can be seen on the lower corner in Figure 5.3.

We created a subscription under ACT_DetectionNode container, which will send a notification to the ACT_LocalService server about the modification in the ACT_DetectionNode Service data. The notification message also contain the modified data. The subscription creation method is show below. In the code, a subscription is created under ACT_DetectionNode, this is subscribed by ACT_LocalService.

```
## subscription
End_point = "https://icon-lab.tim.it/oneM2M/act_test/ACT_DetectionNode"
data1 = """{
    "m2m:sub" : {
      "rn" :  "DetectionNodeNotify",
      "enc" : {"net" : [3]},
      "nu" : ["http://138.96.16.37:5002/LocalService"],
      "nct" : 2}
  }
"""
headers = {'X-M2M-Origin' : 'act_test_prod',
  'X-M2M-RI' : str(int(time.time())),
  'content-Type' : 'application/vnd.oneM2M-res+json;ty=23',
  'Accept':'application/json'}
res = requests.post(url= End_point,
  auth = HTTPBasicAuth('username', 'password'),
  headers = headers,
  data = data1)
res.json()
```

5.4.2 ACT Local Service

For the ACT_LocalService on the ICON platform we created a container the same way we did for ACT Detection Service. On the back end for Local service we have a server running and waiting for the notifications from ACT_DetectionNode. The server is implemented using FLASK library [20]. After receiving the modified data from ACT_DetectionNode ACT Local Service add some information with it and POST it on ICON. The POST method for container and container instance is the same as in ACT Detection Service. The Figure 5.4 will show the ACT Local Service in the resource tree on ICON platform. The brief description of the Local Service server is given below.

```
@api.route('/LocalService', methods=['POST', 'GET'])
def LocalService():
    if(request.method== 'POST'):
```

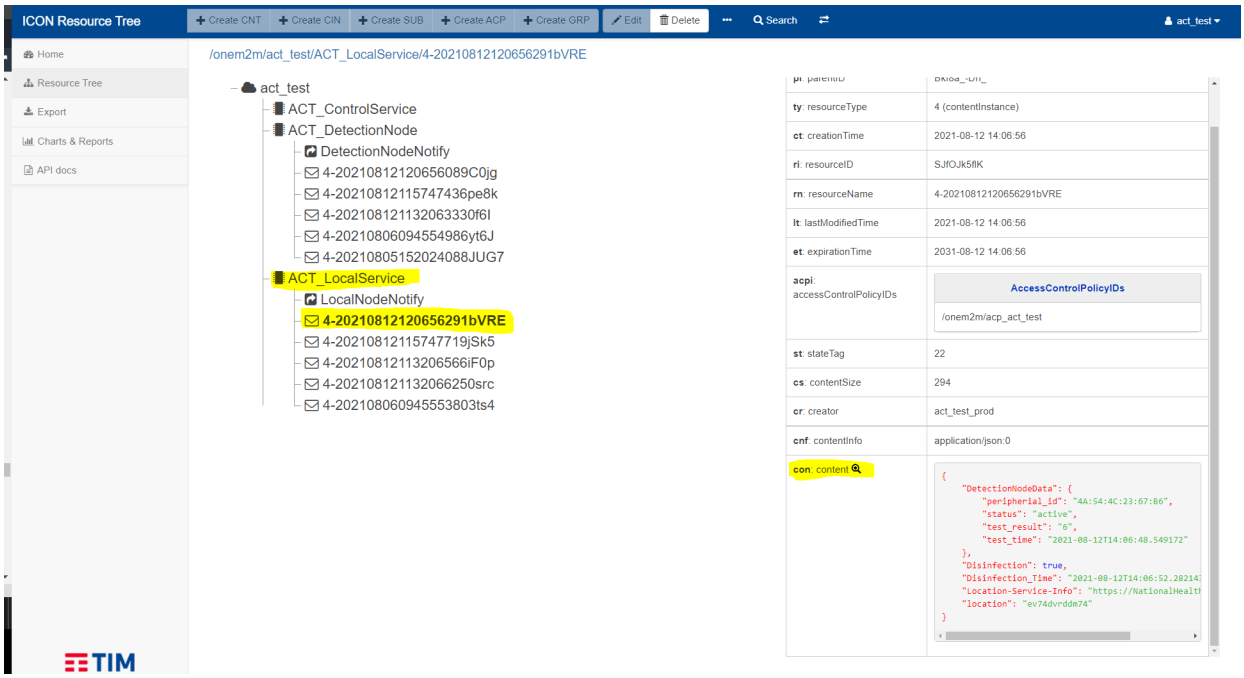

Figure 5.4: ACT Local Service in ICON

```
if request.content_type != 'application/vnd.oneM2M-ntfy+json':
    data= request.get_json()
    return jsonify(data, "ERROR: Not FROM ICON" ), 201
else:
    data = request.get_json()
    LocalServiceData.update(data)
    new_data= data.copy()
    LocalToControlService(new_data)
    return jsonify(LocalServiceData, 201 ), 201
else:
    return jsonify(LocalServiceData)
```

5.4.3 ACT Control Service

For the ACT_ControlService on the ICON platform we subscribed to the notification from ACT_LocalService by created and subscription under ACT_LocalService named "LocalNodeNotify". There are serve on the back end for ACT_ControlService which will add some information to the modified data received and store it in internal database and also on ICON platform. In the ACT_ControlService server we arrange the data

according to the location. It is because when the ACT Smart Mobile Application and ACT Web Application send a query with location. The server take the location parameter from the user as a query and convert it to geo-hash code and then search in internal database for the the data, and send the response in JSON format. The ACT_ControlService on the ICON platform is shown in the Figure 5.5 and the data stored on ICON is shown in Figure 5.6. The ACT_ControlService server on the ICON platform is also written with Flask. The code of the server is given below.

```
@api.route('/ControlService', methods=['POST', 'GET'])
def ControlService():
    if(request.method == 'POST'):
        if request.content_type != 'application/vnd.oneM2M-ntfy+json':
            data = request.get_data()
            return jsonify(result), 201
        else:
            data = request.get_json()
            ControlServiceInfo.append(data)
            new_data = data.copy()
            ControlserviceData(new_data)
            return jsonify(ControlServiceInfo, 201 ), 201
    else:
        return jsonify(result), 200
@api.route('/ControlService/location', methods=['GET'])
def location():
    latitude = request.args.get('latitude', type= float)
    longitude = request.args.get('longitude', type= float)
    get_location_info(latitude, longitude)
    return jsonify(result), 200
```

5.4.4 ACT Display Application

The ACT Display Application is designed with Python. The ACT Display Application is not created on the ICON platform because ACT Display Application is not an oneM2M entity. ACT Display Application will send a query to the ACT_ControlService with the location attributes and will wait for the response form the ACT_ControlService. The

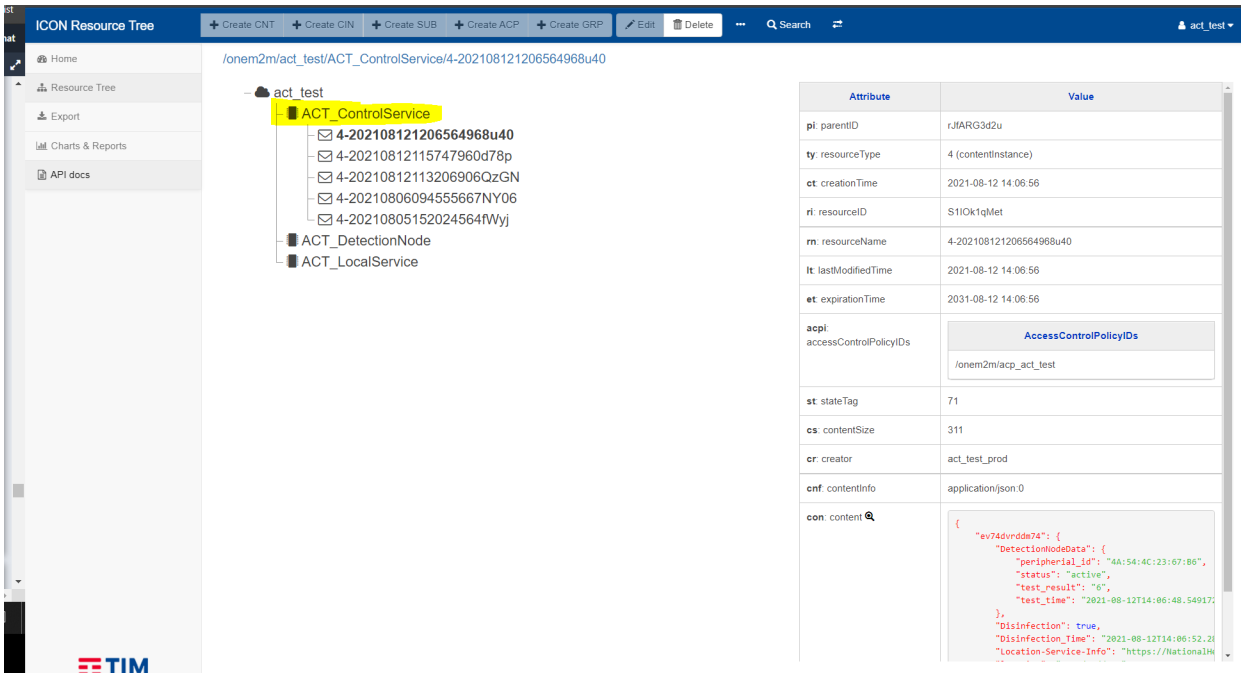

Figure 5.5: ACT_ControlService in ICON

Figure 5.6: JSON Content of ACT_ControlService Container

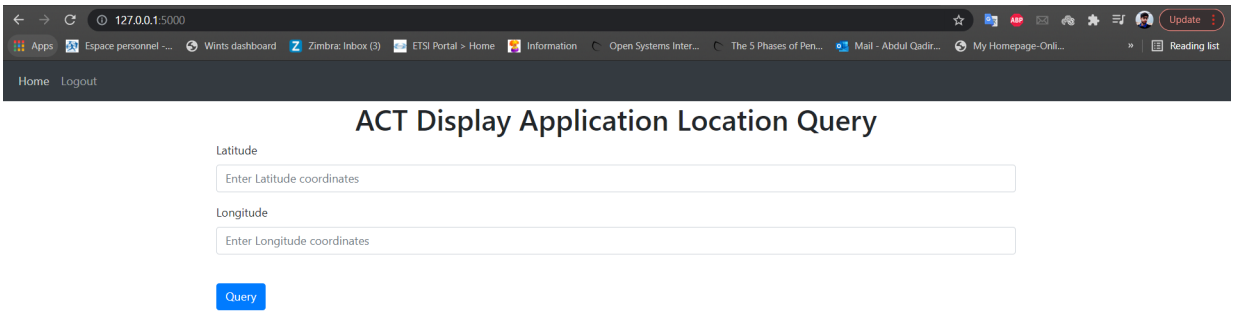

Figure 5.7: ACT Display Application GUI

GUI of the web application is given in Figure 5.7.

5.4.5 ACT Smart Mobile Application

Implementation of an ACT Smart Mobile Application is out of the scope of the document and of the internship. The implementation of ACT Smart Mobile Application is left for future work.

5.5 Experiments

For experiment of ACT implementation with the help of the TIM ICON oneM2M platform are in progress right now. We are trying to simulate an ACT real case.

For experiment we implemented and generated data using random functions. We used RandMac library to generate random BSSIDs. as shown below in the code.

```
def Generate_BSSID(n):
    example_mac = "00:00:00:00:00:00"
    for i in range(n):
        generated_bssid = RandMac(example_mac)
        BSSID.append(generated_bssid)
    return BSSID
```

After getting the BSSIDs we will assign it to the nodes of the graph. The graph in our case will shows a building. The graphical representation is shown just to get some

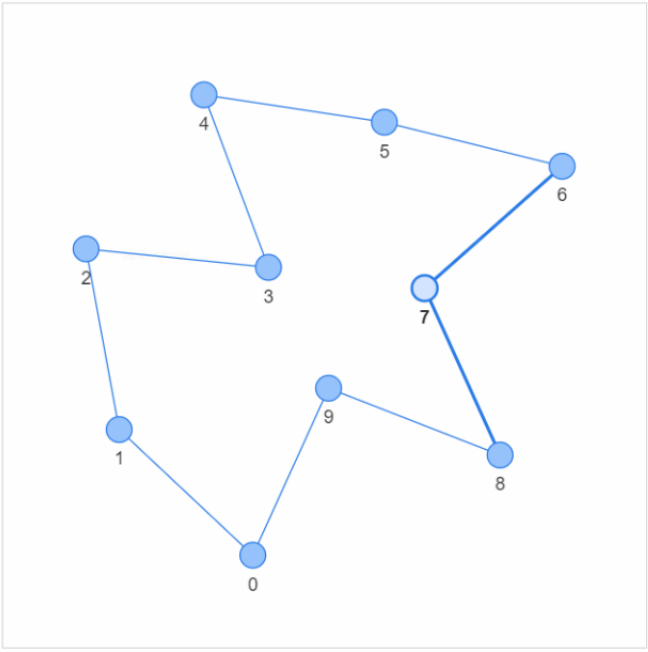

Figure 5.8: Graphical Representation of ACT Use Case

graphical representation. For graph we used used python library NetworkX [21] and Pyvis.network [22]. The code is shown below and the graph is also shown in the figure 5.8. The nodes in the figure shows the corridors and these are assigned the value as BSSID which means each corridor has one ACT Peripheral Device.

```
nx_graph = nx.cycle_graph(n)
for i in range(n):
    nx_graph.nodes[i]['title'] = str(id[i])
nt = Network('500px', '500px')
nt.from_nx(nx_graph)
nt.show('graph.html')
```

After that we we will send a bunch of the ACT Detection Service data to the ICON. For this purpose we used a sleep function inside python which help us to get the different time of the test results. We did a little improvement to the ACT Control Service we made a Mango-DB database on a cloud to store all our received data. It is to be noted that the data is also stored on ICON platform. The reason behind implementing a Mango-DB database is that it will give us more options while implementation of the ACT Smart Application.

On the user side, we will first see the number of corridor the used visit. The Smart

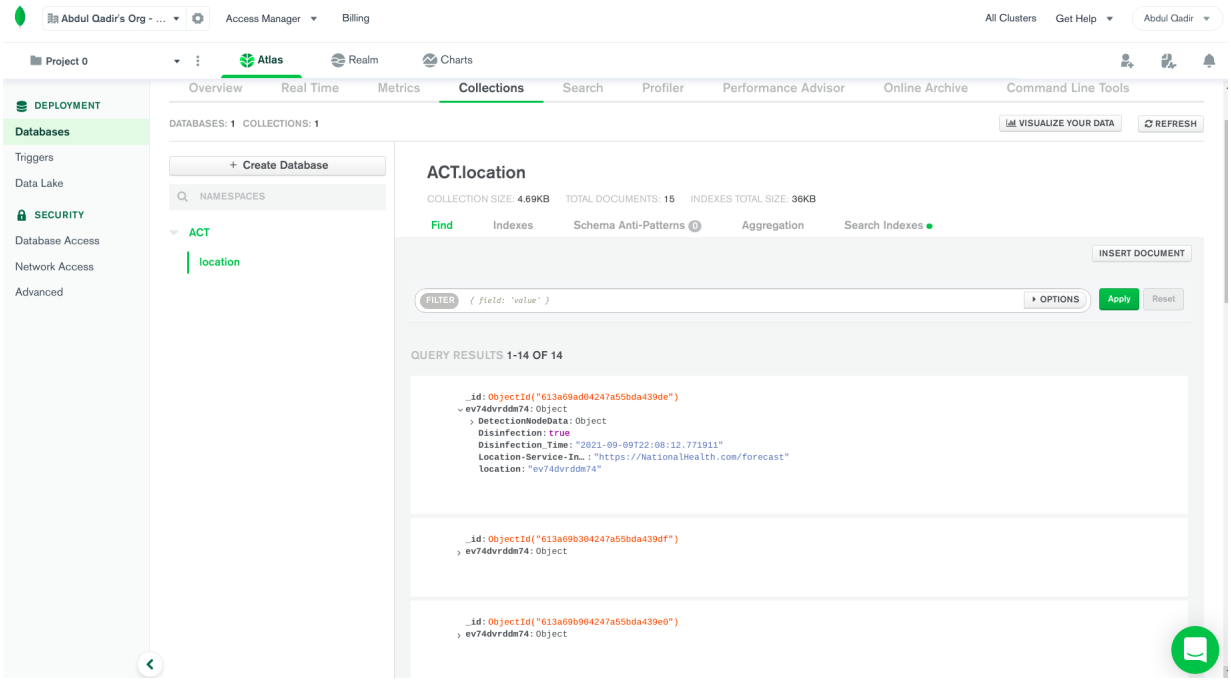

Figure 5.9: Screenshot of Mango-DB

Application will then save the BSSIDs of the ACT Peripheral Service in the corridor.In our experiment we will use Random Walk function of the graphs to get the random sequence of the visited corridors. We used the following random walk script to get a simple random walk for a user. The visited corridors are colored red and the BSSID is stored, which will help us in filtering the results. The random walk on the graph is shown in figure 5.10.

```
def Randwalk(n):
    x = 2
    y = 3
    time = [x]
    position = [y]
    for i in range (1,n+1):
        move = np.random.uniform(1,5)
        if move < 3:
            x += 1
            y += 1
        if move > 3:
            x += 1
```

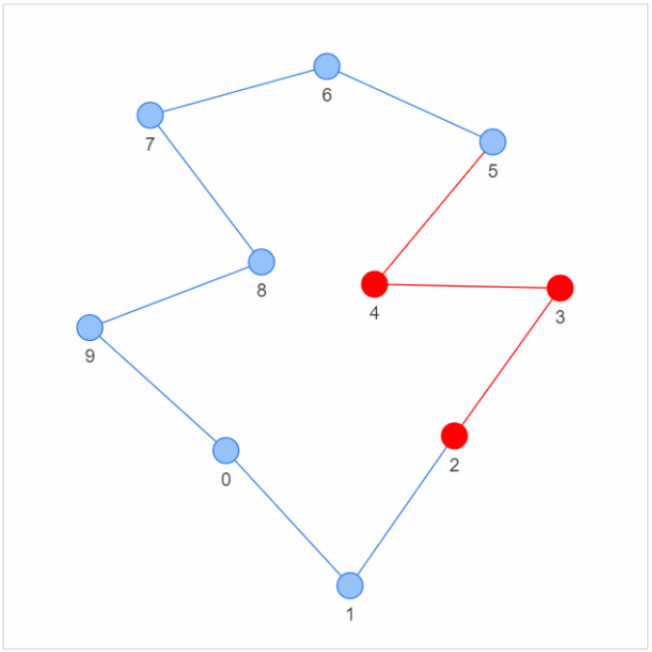

Figure 5.10: Random Walk of User

```
            y += -1
        time.append(x)
        position.append(y)
    return [time, position]
```

For query as the user will send his location attributes in query to get the information related to that location. In our implementation we can do query from a simple python code using Requests. Also, we can get the response to the query from ICON platform. So on the user side until now we do a simple query using request. the query mechanism will be enhanced and will be implemented with the ACT Smart Application. The query is shown below:

```
# api-endpoint
URL = "http://138.96.16.37:443/ControlService/location"
PARAMS = {'location':'ev74dvrddm74'}
r = requests.get(url = URL, params = PARAMS)
# extracting data in json format
data = r.json()
print(data)
```

The response from the server is shown in the figure 5.11.

```
In [9]:
# api-endpoint
URL = "http://138.96.16.37:443/ControlService/location"

PARAMS = {'location':'ev74dvrddm74'}

r = requests.get(url = URL, params = PARAMS)

# extracting data in json format
data = r.json()
print(data)
```

```
{"location_data": {"_id": {"$oid": "613a6e05bb982cbcebf708b4"}, "DetectionNodeData": {"peripherial_id": "4A:54:4C:23:67:B6", "s
tatus": "active", "test_result": "5", "test_time": "2021-09-09T22:26:42.131090"}, "Disinfection": true, "Disinfection_Time": "2
021-09-09T22:26:45.098924", "Location-Service-Info": "https://NationalHealth.com/forecast", "location": "ev74dvrddm74"}}
```

Figure 5.11: Query Response from ACT Control Service

Once in the near future we have the completely implemented ACT Smart Application and ACT Web Application we will try to simulate the complete use case. We have defined some key performance indices by the help of which we will evaluate the protocol state and simulation state. Below we will present some of the possible KPIs.

- Density of customers per time frame Vs frequency of tests
- Number of infected spots registered by ACT Smart Application Vs frequency of tests
- Time duration between test result and test sample
- Length of path of infected user
- Global exposure time to ACT access point Vs given time frame
- Intersection of transmission range of Access Point Vs deployed topology
- Number of queries per hour with smart phone (location)
- Number of queries per hour with web application (location, forecast color and forecast frame)
- Time between a web query and physical travel to the ACT installed location

5.6 Future Tasks

The implementation of the ACT Smart Mobile Application and enhancement of the ACT Web Application are future task of this project. Once completed then a real proof of concept implementation and deployment of ACT in a real life use case experiment also can be envisageable.

Chapter 6

Conclusions

This document provide the theoretical and implementation details of a Asynchronous Contact Tracing ETSI standard protocol. This protocol is standardized by the ETSI SmartM2M and oneM2M Technical Committee. The main focus of the protocol is to provide protection against COVID-19 pandemic. Also the protocol can be extended and customized to be used as protection against any kind of future pandemics. ACT overcome the time and distance constraint of the all present Digital Contact Tracing mechanisms/protocols. ACT enables people who have come into contact asynchronously with those particular materials to be alerted of a potential COVID-19 contagion, and, at the same time, it signals that one or more persons have been in contact with the material which is now spreading the SARS-CoV-2 virus. ACT is particularly effective, considering that the SARS-CoV-2 virus can survive for a significant time on certain materials. The level of contamination depends on the nature of the surface the concentration of the virus, the ambient temperature, the season of the year, the level of humidity, and exposure to sun light. The period of contamination can span from a few hours to several days.

ACT tests materials that have been in contact with humans and uses wireless and IoT technologies to notify a particular contamination event to Public Health Authorities that, in turn, will inform users who were potentially close to those infected areas. The ACT goal is to is to trace the material hosting virus and inform the people in surroundings. It can be applied in any place where people share the same physical space, such as a supermarket, schools, restaurants, hotels, gyms, offices, working plants, hospitals etc. It can also be applied to objects in movement, such as a bus/metro in a transport network. The methodology be used as a recommendation tool for the Public Health Authority to impose a selective lockdown instead of full lockdown.

The ACT ETSI standard will promote individual testing in the event of the user receiving a notifications that he/she is potentially in danger. Moreover, it does not require the transmission of any personal information by the user, respecting both EU GDPR and people very sensible to personal privacy.

ACT uses ETSI oneM2M infrastructure for its implementation. An ACT implementation has been done with the help of the ICON Platform developed by Telecom Italia Mobile. ICON is an ETSI oneM2M based middleware implementation platform provided by Telecom Italia Mobile (TIM). ICON make use of REST architecture which allow tenant to Read, Write, Update, and Delete data from ICON. We used ICON for implementation of ACT in oneM2M domain. The Python implementation of the different entities of the ACT is also shown.

ACT can be simulated for real case scenario in future. The update of the implementation is also possible. A set of experiments can be done for a use case in future.

Bibliography

[1] L. Liquori, S. Wood, P. Guillemin, and E. Scarrone, "Asynchronous contact tracing." [Online]. Available: https://hal.inria.fr/hal-02989404/file/Academia_Asynchronous_Contact_Tracing%201.1.pdf

[2] "onem2m homepage." [Online]. Available: https://www.onem2m.org/

[3] C. Castelluccia, N. Bielova, A. Boutet, M. Cunche, C. Lauradoux, D. Le Métayer, and V. Roca, "Robert: Robust and privacy-preserving proximity tracing," 2020.

[4] "Pan-european privacy-preserving proximity tracing (pepp-pt)." [Online]. Available: https://github.com/pepp-pt/pepp-pt-documentation

[5] C. Troncoso, M. Payer, J.-P. Hubaux, M. Salathé, J. Larus, E. Bugnion, W. Lueks, T. Stadler, A. Pyrgelis, D. Antonioli *et al.*, "Dp3t: Decentralized privacy-preserving proximity tracing," *École polytechnique fédérale de Lausanne and ETH Zurich. Retrieved September*, vol. 1, p. 2020, 2020.

[6] "Exposure notification system." [Online]. Available: https://covid19.apple.com/contacttracing

[7] "Asynchronous contact tracing system fighting pandemic disease with internet of things (iot)." [Online]. Available: https://www.etsi.org/deliver/etsi_ts/103700_103799/103757/02.01.01_60/ts_103757v020101p.pdf

[8] S. J. Olsen, H.-L. Chang, T. Y.-Y. Cheung, A. F.-Y. Tang, T. L. Fisk, S. P.-L. Ooi, H.-W. Kuo, D. D.-S. Jiang, K.-T. Chen, J. Lando *et al.*, "Transmission of the severe acute respiratory syndrome on aircraft," *New England Journal of Medicine*, vol. 349, no. 25, pp. 2416–2422, 2003.

[9] A. P. Harvey, E. R. Fuhrmeister, M. E. Cantrell, A. K. Pitol, J. M. Swarthout, J. E. Powers, M. L. Nadimpalli, T. R. Julian, and A. J. Pickering, "Longitudinal monitoring of sars-cov-2 rna on high-touch surfaces in a community setting," *Environmental Science & Technology Letters*, vol. 8, no. 2, pp. 168–175, 2020.

[10] K. M. O'Reilly, D. J. Allen, P. Fine, and H. Asghar, "The challenges of informative wastewater sampling for sars-cov-2 must be met: lessons from polio eradication," *The Lancet Microbe*, vol. 1, no. 5, pp. e189–e190, 2020.

[11] G. Kampf, D. Todt, S. Pfaender, and E. Steinmann, "Persistence of coronaviruses on inanimate surfaces and their inactivation with biocidal agents," *Journal of hospital infection*, vol. 104, no. 3, pp. 246–251, 2020.

[12] R. Dorfman, "The detection of defective members of large populations," *The Annals of Mathematical Statistics*, vol. 14, no. 4, pp. 436–440, 1943.

[13] "Gdpr rules." [Online]. Available: https://gdpr-info.eu/

[14] "Iso 8601: Date and time format." [Online]. Available: https://www.iso.org/iso-8601-date-and-time-format.html

[15] I. C. S. L. S. Committee *et al.*, "Ieee standard for information technology-telecommunications and information exchange between systems-local and metropolitan area networks-specific requirements part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications," *IEEE Std 802.11ˆ*, 2007.

[16] "Geo-hash code system." [Online]. Available: http://geohash.org/

[17] "Functional architecture onem2m." [Online]. Available: https://www.onem2m.org/technical/published-specifications/release-1

[18] "Requests in python." [Online]. Available: https://docs.python-requests.org/en/master/

[19] "Geo-hash library." [Online]. Available: https://pypi.org/project/pygeohash/

[20] "Flask." [Online]. Available: https://flask.palletsprojects.com/en/2.0.x/

[21] "Networkx." [Online]. Available: https://networkx.org/

[22] "Pyvis." [Online]. Available: https://pyvis.readthedocs.io/en/latest/tutorial.html

Appendices

Appendix A

JSON Representation of ACT Messages

A.1 Information Exchanged Between Detection Service and Local Service

```
ACTDetectionServiceToACTLocalServiceMessage:
{
  "Status": "Active",
  "Test-Time": "2021-07-08T11:28:27.176750",
  "Test-Result": 3
}

ACTLocalServiceToACTDetectionServiceMessage:
{
  "Command": "Restart",
  "Test-Interval": 3
}
```

A.2 Information Exchanged Between Local Service and Peripheral Service

```
{
  "Peripheral-Service-Id": "C8:60:00:4C:27:A5"
```

```
}
```

A.3 Information Exchanged Between Local Service and National Control Service

```
ACTLocalToACTNCServiceMessage:
{
  "Peripheral-Service-Id": "C8:60:00:4C:27:A5",
  "Location": "ezs42e44yx96",
  "Location-Service-Info": "https://NationalHealth.com/forecast",
  "Test-Result": 3,
  "Test-Time": "2021-07-08T11:28:27.176750",
  "Disinfection": true,
  "Disinfection-Time": "2021-07-08T11:28:27.176750"
}
```

A.4 Information Exchanged Between Smart Mobile Application and National Control Service

```
ACTSmartAppToACTNCServiceQuery:
{
  "ACTSmartAppToACTNCServiceQuery": "ezs42e44yx96"
}
ACTNCServiceToACTSmartAppQueryResponse:
{
  "Replies": {
    "Peripheral-Service-Id": "C8:60:00:4C:27:A5",
    "RED-Forecasts": {
      "start": "2021-07-08T11:28:27.176750",
      "end": "2021-07-08T11:28:27.176750"
    },
    "Local-Service-Info": "https://LocalHealth.com/forecast",
```

```
    "Control-Service-Info": "https://NationalHealth.com/forecast"
  },
  "Message": "The area is infected, please get
  ASAP a PCR test or go to isolation"
}
```

ACTSmartAppToACTNCServiceMessage:

```
{
  "Personal-Test-Time": "2021-07-08T11:28:27.176750",
  "Personal-Test-Code": "T23569K34",
  "Visited-BSSIDS": [
    {
      "Peripheral-Service-Id": "C8:60:00:4C:27:A5"
    }
  ]
}
```

A.5 Information Exchanged Between Display Application and National Control Service

ACTDisplayApplicationToACTNCServiceQuery:

```
{
  "Query": "ezs42e44yx96",
  "Forecast-Frame": {
    "start": "2021-07-08T11:28:27.176750",
    "end": "2021-07-08T11:28:27.176750",
    "Forecast": "RED"
  }
}
```

ACTNCServiceToACTDisplayApplicationqueryResponse:

```
{
  "Replies": {
    "Detection-Area": "ezs42e44yx96",
    "Forecasts": [
      {
        "Forecast-Color": "Green"
      }
    ],
    "Forecast": {
      "Forecast-Color": "Red",
      "Frame": {
        "start": "2021-07-08T11:28:27.176750",
        "end": "2021-07-08T11:28:27.176750"
      }
    }
  },
  "Message": "Message from National Health Service"
}
```


A.6 Information Exchanged Between different National Control Services

ACTNCServiceToACTNCServiceQuery:


```
{
  "ACTNCServiceTOACTNCServiceQuery": "ezs42e44yx96"
}
```


ACTNCServiceToACTNCServiceQueryResponse:


```
{
  "Replies": {
    "Peripheral-Service-Id": "C8:60:00:4C:27:A5",
    "Detection-Area": "ezs42e44yx96",
```

```
    "Forecasts": [
      {
        "Forecast-Color": "Green"
      }
    ],
    "Forecast": {
      "Forecast-Color": "Red",
      "Frame": {
        "start": "2021-07-08T11:28:27.176750",
        "end": "2021-07-08T11:28:27.176750"
      }
    }
  },
  "Message": "Message from National Health Service"
}
```

Appendix B

Source Code

B.1 Source Code

The source code of this protocol can be find Source Code. The update and last version of the code and experiments will be uploaded on 15th September.