2020

University of North Carolina Wilmington

Master of Science in

Computer Science and Information Systems

Proceedings

https://csbapp.uncw.edu/mscsis

USING KNOWLEDGE GRAPH FOR ACCESS CONTROL IN SMART HOME ENVIRONMENT

Deepthi Thalagundamatada

A Capstone Project Submitted to the
University of North Carolina Wilmington in Partial Fulfillment
of the Requirements for the Degree of
Master of Science

Department of Computer Science
Department of Information Systems and Operations Management

University of North Carolina Wilmington

2019-2020

Approved by

Advisory Committee

Dr. Hyunbum Kim _____

Dr. Minoo Modaresnezhad _____

_____
Dr. Sudip Mittal, Chair

Accepted By

_____
Dean, Graduate School

TABLE OF CONTENTS

FIGURES:

TABLES:

ABSTRACT

Using knowledge Graph for Access Control in Smart Home Environment

Deepthi Thalagundamatada, 2019. Capstone Paper, University of North Carolina Wilmington.

A 'Smart Home' installation consists of many Internet of Thing (IoT) devices deployed inside the house. An IoT device can have multiple sensors that collect and transfer data over a network to various external servers. However, these devices collect a lot of sensitive data about various residents living in a house. This data is sent to various external cloud applications that can store this information.

To give residents better control over their privacy, a semantically rich knowledge graph driven system that has been built can evaluate different packets and 'drop' them as per the 'policy' set in place by users/residents. These policies will have rules in place for different IoT devices installed in a smart home environment. For example, a policy can state that an Alexa should not communicate with the internet from 10 pm to 6 am. Access control system implemented has the capabilities of asserting the captured packet in a knowledge graph, Evaluate if the transmission of the packet is 'allowed' as per the policy set by the resident user.

This access control system which is built has been accomplished by using different technologies and tools like Knowledge Graphs, Protégé, Semantic Web Rule Language (SWRL).

CHAPTER 1: INTRODUCTION

It is possible to have the right access control for the smart home environment, by analyzing the packets captured at the router level which are further evaluated from the assertion of a packet into the knowledge graph and resulting in allowing or denying the packet according to the policies defined by the user.

The Smart Home consists of a lot of different smart and intelligent devices that consist of actuators, sensors, displays, and computational elements helping people to have a better and sophisticated life by automating the small things which need human interaction by identifying the pattern of the person's life and act accordingly. People are getting addicted and adapting to these smart and intelligent devices quickly and depending on them to lead their life. The privacy and the security of the people who are adapting to these smart end, intelligent devices must be considered seriously, smart devices without the right security controls in place can be dangerous. Though these devices make life easier and hassle-free, we must have the right access controls in place to avoid unnecessary attacks from the intruders by using these smart devices. Access control can help to avoid the attacks that can take place.

Smart homes have got a wide range of scope and limelight on it due to its flexibility of integration into the pace of life smartly by adjusting itself according to the behavior of the people around it. The sensing technologies are used to track the activities of the people that they perform and communicate with the physical devices to co-operate the daily activities of the people in the house in a better manner. These physical devices are usually

infrastructure appliances which are widely named as IoT devices or the smart devices which are combined in the smart environment and leverage network to communicate to the application or the server to carry out the operations [1].

Many devices or sensor networks communicate with each other are embedded through servers in fixed infrastructure. These can then be managed by a person in the smart home environment. These devices integrate computation, networking, and physical process which can be monitored by various menu settings for the device. This can improve the privacy and security of the people managing a smart home environment. It is essential to collect the data from the sensor and communicate it to the applications over the network so that the smart device's data can be analyzed, and intelligent decisions can be made using that data. [1]

Network-level security can be implemented for all IoT devices, instead of device-level security is specific to a device; Network-level security enhances the protection that can strengthen any device-level security implemented by the manufacturer [2].

In smart home applications, having a flexible access control is very important so that users can transfer or alter the permission as per their needs. The right access control and permission need to be in place for both smart devices and the people who operate them [3].

As smart home appliances are enabled with different types of sensors, different technologies like smart camera, energy consumption sensors, temperature, humidity and face recognition, these connected devices generate a huge amount of sensitive data which

can be cross verified, interpreted by applying specialized rules in such a way to make the related action.

To have control over the devices we can embed them with the 3 levels of rules [1] -

(i) Rules for managing correlating sensors data and technologies [1].

(ii) Rules able to guide the smart home process, which defines the users and system behavior, to protect against the possible problems and inconvenience faults [1].

(iii) Rules that manage access to the building, specific rooms, tools, or data to protect against malicious usage or security flaws. [1].

Despite a high level of heterogeneity because of various communication media and network protocols, the smart home systems are integrated into a well laid structured network which is called a smart home internal network. Integration achieved using a central node called residential gateway (RG) which serves as the bridge between the internal network of the smart home environment and the internet world. The security in the smart home environment relies on the six essential properties which need to be maintained properly; confidentiality, integrity, authentication, authorization, non-repudiation, and availability [4][6].

This project entitles to have the right access controls in place for the devices so that the devices do not communicate unnecessarily by using the policies set by the user. Every device inside the home network needs to connect to the internet which goes through the router/internet gateway. Internet gateway acts as a barrier to these devices to connect to the outer world or the internet. The idea of having the policies which define access control at

the gateway level to the device can help to decide that it can allow packets from the home network to the internet or the applications on the cloud or the other networks. The household intelligent devices need to have this kind of access control policies in place and updated by the user to have better privacy.

The important thing to note is that the policies to be set by the user or the people who are living in the house. For example, a user can define the policy as Alexa in the house should not send the data in the midnight to the server outside the house. The Smart door should not open after the hours which is set by the user so that no one enters the house later the time without the house owner intervention.

These policies are set for the devices, this can be done having the static IP addresses assigned to the devices rather than the DHCP. The static IP address should be defined for the devices to identify the device for which the policy is defined. Using static IP addresses assigned to the devices we can identify the devices and apply the policies defined by the user to allow or deny the packets from the devices which also enable better controls in place for the devices inside the house.

The rest of this document is organized as follows –Section 2- describes the related work which is divided into sections which describe the IoT devices, knowledge graph, and the last section describes the relation between IoT devices and the knowledge graph. Section 3- contain the methodology which describes system architecture and components involved.

CHAPTER 2: RELATED WORK AND BACKGROUND

2.1 IOT DEVICES

IoT devices are more often resource-constrained and deployed in an unmonitored and unsecured environment. IoT devices need the right authentication and authorization with the right access control over the devices and people, this enables the security of the IoT devices and the information exchanged from the IoT devices. Many authentications and authorization schemes are proposed but there is a need for attention towards the access control mechanisms or the process which needs to be in place to have better protection from insider threats [5].

Network-level security can be implemented across the various range of IoT devices which can lead to device-level security which is specific to the device. Device-level security enables to have the security which is embedded into the devices and network-level security at the gateway of the home network is more essentially needed to make the secure internal home network for the IoT devices. [6]

The IoT devices at the gateway level can be monitored easily and by defining simple policies at this level the intrusion needs to be analyzed and updated timely manner like a password that enables a more secure system in the smart home environment. Consequences of false authorization or access can lead to a certain capability to the devices which can be misused. The default policies at the router level should be changed and default should be assigned to the right device and people. False default authorization can be a loophole of

the security system. Default settings must be determined to mitigate the intrusion by choosing the likely candidate. This paper revokes the thinking to have the right defaults in pace for the right person [7].

The [Distributed Denial of service] DDOS attacks (The attack on the IoT device where they bring down the (domain name server) DNS down) in the recent past has been known to everyone and there are serval attacks on the smart home devices too which is enabling users to think of establishing the IoT devices at the house and to have the smart home environment. The necessary steps need to be taken at the user end to mitigate the issues because there are serval examples that happened due to privacy violations.

Data that are aggregated from the IoT devices can be of 2 types -

- Behavioral data
- Network data.

The status of the sensors like switch off/on and the sensor readings can be considered as the behavioral data. Behavioral data would describe the state of the device. Using API calls from various cloud repositories we can get behavioral data.

Network data is the data which is obtained from IoT devices through network protocols like TCP/UDP packets. The integration of both the data (behavioral data and network data) is necessary to study the smart home environment. [8].

2.2 KNOWLEDGE GRAPH

Definition of the Knowledge Graph:

“A knowledge graph-

1. Mainly describes real-world entities and their inter-relations, organized in a graph
2. Defines possible classes and relations of entities in a schema.
3. Allows for potentially interrelating arbitrary entities with each other.
4. Covers various topical domains.” [9].

Representing the things and the connection between things and the data related to them enhance the knowledge which can be effectively represented in the form of the graph can be referred to as a knowledge graph. Using a knowledge graph, we can distinguish between the real-world things and data related and connect the things which enable us to fetch the more meaning full data from the graph.

The huge integrated collection of data and information formed with the help of huge interconnected links between the data. The meaning full insight from the huge data lake can be obtained by representing data as a concept, things, people, etc. by using the graph fills up the gap of connection between them.

Knowledge graphs and its usage is becoming extensively used in many applications and arenas. Google uses the knowledge graph to distinguish the search made by the user and provide insight related to it. For example, when the user searches for some company or the person in google, google relates the information using a knowledge graph and it runs in the

knowledge graph and displays the result of the search is as a box which is also known as the knowledge panel. Fig 1 shows the knowledge panel generated as an output for the search query ‘IBM’ and the information which we get from google on the right-hand side of the search page in the box [knowledge panel] which displays the brief description, CEO, subsidiaries, etc. These are related to IBM and are generally stored in a knowledge graph. The more interesting thing is that we can even look at what other people looked out for when they did the similar kind of search, this also links to the profile of the company and the section where people also searched for is also fetched from the knowledge graph. Google uses a wiki and many others as the sources to construct this kind of graph of knowledge and display it a format where it briefs about the search.

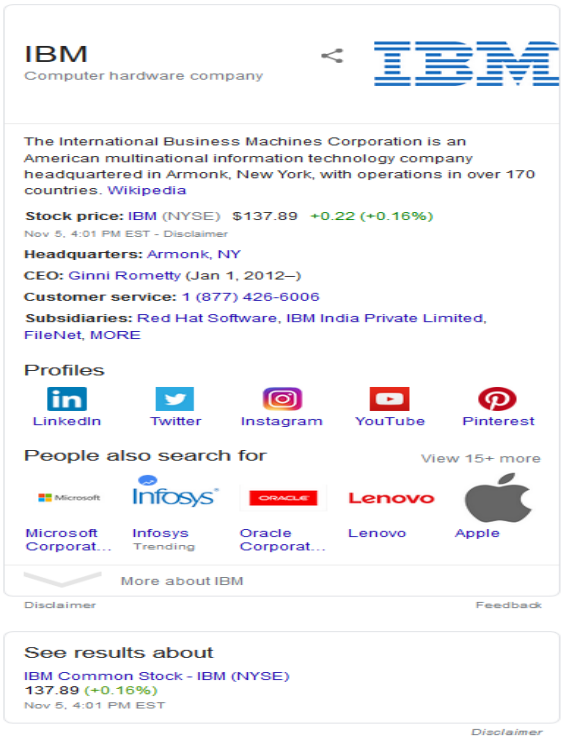

Figure 1: Google Knowledge Graph

Figure 2: LinkedIn knowledge graph

LinkedIn is using a knowledge graph to match up the skill of the member in search and identify what skill set they need to have an open-ended market of the job, by which the job market gap. We can see the knowledge graph of LinkedIn in the above picture.

IBM has extended the knowledge graph usage to the new level. From IBM research Ireland the explain that they are doing extensive research to connect the AI and understand the IoT data in new ways by combining the machine learning with the knowledge graph to have an extensively more intelligent and analytical smart system, by forming the knowledge graph of IoT which can interrelate and can semantically segregate the things and data to autonomously analyze and understand the life cycle data and arrive at the right decision. Information on knowledge graph can be represented in the formal ontology and semantics which can be easily available for automating information processing. data modeling statements in ontological language and world of knowledge graphs for data are incremental in every aspect. We can easily modify the concept by enhancing or modifying the data model.

2.3 IOT AND KNOWLEDGE GRAPH

Using a semantically rich access control system that accommodates an access broker module to evaluate access decisions based on rules generated using the policies declared by the user and ensures to follow the updates are implemented accordingly. [10]

IoT introduces quite a bit of new challenges in terms of processing and storage of large quantity of the data which are produced by the edge analytics network.

"The key goals of edge analytics framework are

(1) Simplifying the integration of heterogeneous hardware and software resources with the existing applications.

(2) Using knowledge graph for reusability, relationship interfacing, maintainability, and data communication normalization for infrastructure normalization." This paper addresses these challenges. [8]

The knowledge graph is a centralized ontology that contains information regarding the schema where each edge device type and the capabilities and data sources that bind them, this can be used as the foundation for the knowledge graph. knowledge graph also contains the instances related to the applications, these describe the current ecosystem within the represented network. [11]

The ontology deals with the description and observations of the resource of multiple heterogeneous IoT deployments. Ontology is a rich and complex knowledge that is represented in the form of the graph nodes, expressing the things and links between nodes that define the relationship between two things. Things can be classified into class and instances of the superclass of that class. Classes and relationships from one or more

taxonomies. Each class and different properties need to be defined with a unique identified namespace prefix and the class or property with a specific name.[6]

IoT middleware is the software layer that enables the IoT Devices to be interconnected to provide intelligent decisions. Machine to Machine access plays a very important role in the IoT middleware which is called heterogeneous access. Knowledge graphs are used to specify the terms and relations at the top level. The instances are considered as data in the knowledge graph. IoT knowledge graph consists of three parts IoT base ontology, IoT Domain Ontology, IoT instance. [12]

The Graph of Things(GoT) is a link of heterogeneous IoT data source using Linked Data which is a scalable and elastic software stack which can deal with billions of records of historical datasets in conjunction with millions of triples being fetched and enriched to connect to GoT per hour in real-time. This graph enables the smarter way to discover and explore the IoT data under meaningful facts and relationships. [13]

Using web ontology language (OWL) [14], the policies can be represented formally. It can be used to represent security policies [15] [16]. Complex ontologies can be represented using OWL effectively. Owl can also be used to represent ABAC policies [15]. OWL classes can be used to define the basic constructs. object attribute, subject attribute, user attribute can be defined in terms of OWL properties which can be used in SWRL which has the syntax In this syntax, a rule has the form: antecedent ⇒ consequent. [11]

2.3.1 IOT ONTOLOGY LITE

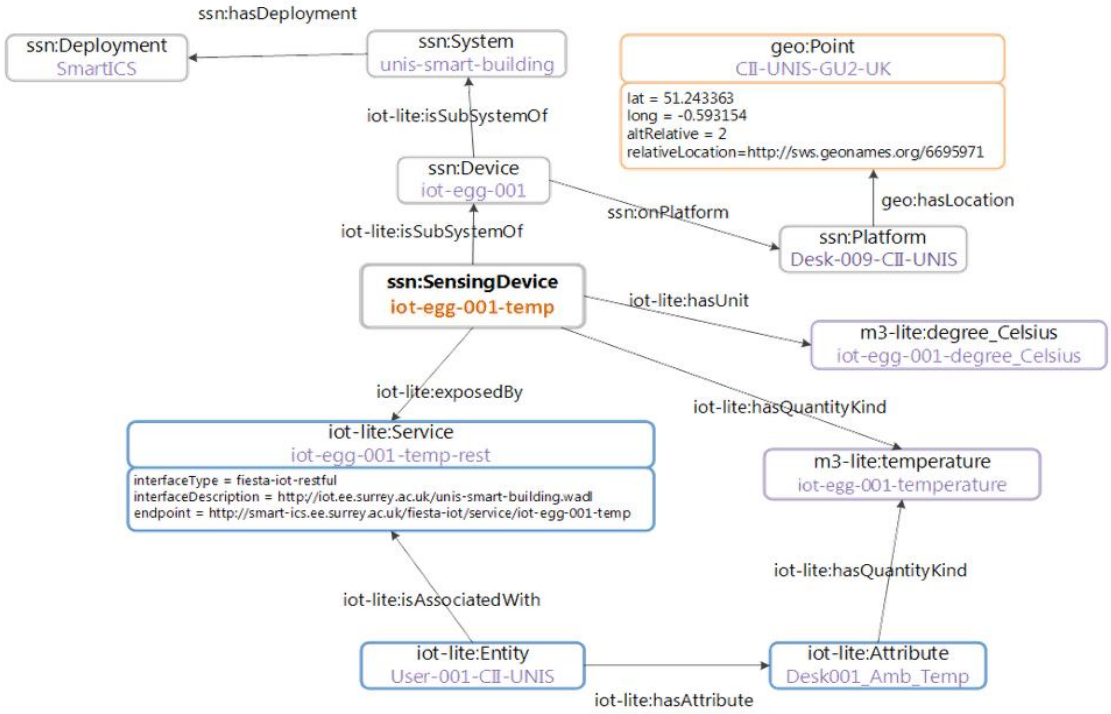

Figure 3: IoT Lite

IoT involves more Machine to machine communication than the human. Most of the communications and information processing would be from IoT and can be seen all over the place. IoT ontology lite is a kind of semantics to represent the IoT communications, It would simplify the complex models and could help to describe the overall system by defining the main concept of the IoT domain. [17]

Using IoT lite one can describe IoT Concept in three classes:

- Objects: Object is the IoT entity (i.e. room, car, table)
- System resources: A system is a unit of abstraction for the piece of infrastructure for sensing. A system has components, its subsystems which can be other systems
- Services: This refers to the service rendered by the IoT device. [17]

IoT devices can be represented in three classes:

- Sensing devices: The Device which collects the sensor data which are measurable and can be analyzed.
- Actuating devices: The Device which is used for operating, which provides the action to perform
- Tag devices: Device that can redirect to a resource with information.

IoT environments are dynamic and they need to have interoperability at the same time which is necessary to be considered in the IoT model. This concept of having both dynamic and interoperability together needs the expressiveness versus complexity is a challenge.

CHAPTER 3: TOOLS AND TECHNOLOGIES USED

3.1 SWRL

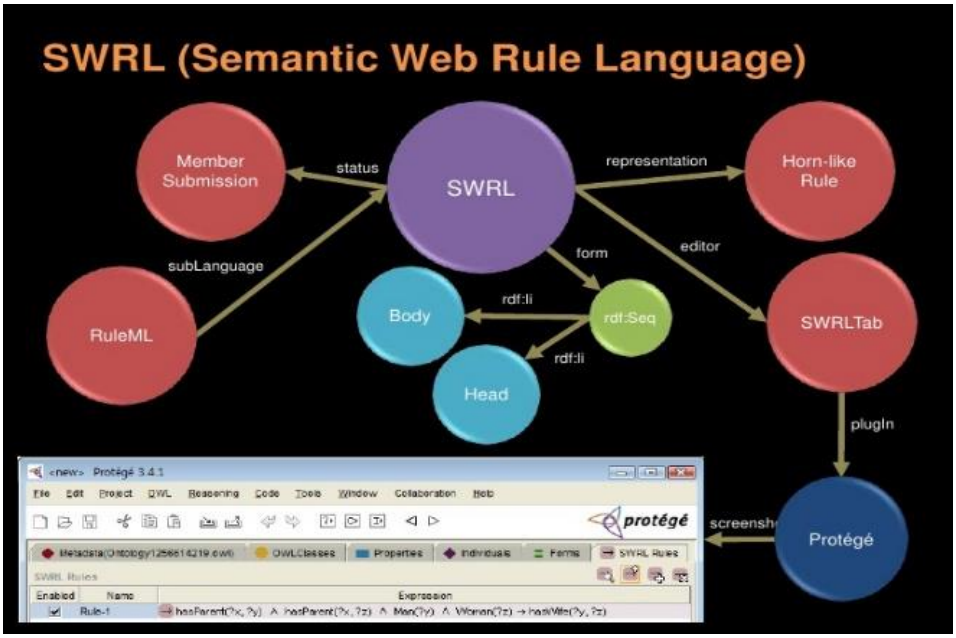

Figure 4: SWRL

SWRL is an acronym for Semantic Web Rule Language. SWRL is intended to be the rule language of the Semantic Web. Policies are represented in SWRL [18]

All rules are expressed in terms of OWL concepts

- Classes
- Properties
- Individuals

The Syntax of SWRL is often Human readable.

In this syntax, a rule has the form:

ANTECEDENT ⇒ CONSEQUENT

Using this syntax, a rule asserting that the composition of parent and brother properties imply the uncle property would be written:

$\mathbf{parent}(?\mathbf{x}, ?\mathbf{y}) \wedge \mathbf{brother}(\mathbf{y}, \mathbf{z}) \rightarrow \mathbf{uncle}(?\mathbf{x}, ?\mathbf{z})$ [18]

Here

Both ascendant and consequent are conjunctions of atoms written as

$a_1 \wedge ... \wedge a_n$.

Variables are indicated using the standard convention of prefixing them with a question mark (e.g., ?x).

The below table can be used for interpreting the rules [18]

Table 1: Interpretation table SWRL

Interpretation Conditions Table	
Atom	**Condition on Interpretation**
C(x)	S(x) ∈ EC(C)
D(z)	S(z) ∈ EC(D)
P(x,y)	<S(x),S(y)> ∈ ER(P)
Q(x,z)	<S(x),L(z)> ∈ ER(Q)
sameAs(x,y)	S(x) = S(y)
differentFrom(x,y)	S(x) ≠ S(y)
builtIn(r,z1,...,zn)	<S(z1),...,S(zn)> ∈ D(f)

Table 2 would help define the rules with the constraints. I have taken this from the user manual of SWRL which I used to write the SWRL rules. [19]

Table 2: SWRL queries with constraint

Interpretation Conditions Table		
Query	**Bindings returned**	**Constraints returned**
`swrlb:equal(?x,3)`	?x = 3	none
`swrlb:notEqual`	?x = ?Var0	?Var0=\=3
`swrlb:lessThan(?x,3)∧ swrlb:lessThan(?y,?x)`	?x = Var0, ?y = ?Var1	Var0<3.0, ?Var1-?Var0<0.0
`swrlb:lessThanOrEqual(?x,3)∧ swrlb:greaterThanOrEqual`	?x = 3	none

3.2 PROTÉGÉ TOOL:

Protégé is a free open source ontology editor that I have used to build the ontology. This has a good and usable user interface. This graphic user-based interface is developed by Stanford and can be used to validate the ontologies based on the rules defined using the SWRL, OWL, etc. This knowledge-based application is a constructive suite of tools that can be used to construct the domain models. This ontology editor can be really helpful to build the POC of the projects which can be used for several things.

It has several tabs that can be utilized wisely to construct the ontology. Mainly I have used entities and individuals by class, SWRL, ontoGraf, OWLViz are used int his Project to build the knowledge-based ontology.

The OWL viz and onto Graf tabs help view the ontology built graphically. SWRL tab is used to build the SWRL rules and run the rules on the ontology built and Entities and individual by class tabs are used to add the entities and define the like object property, data property, etc. [20]

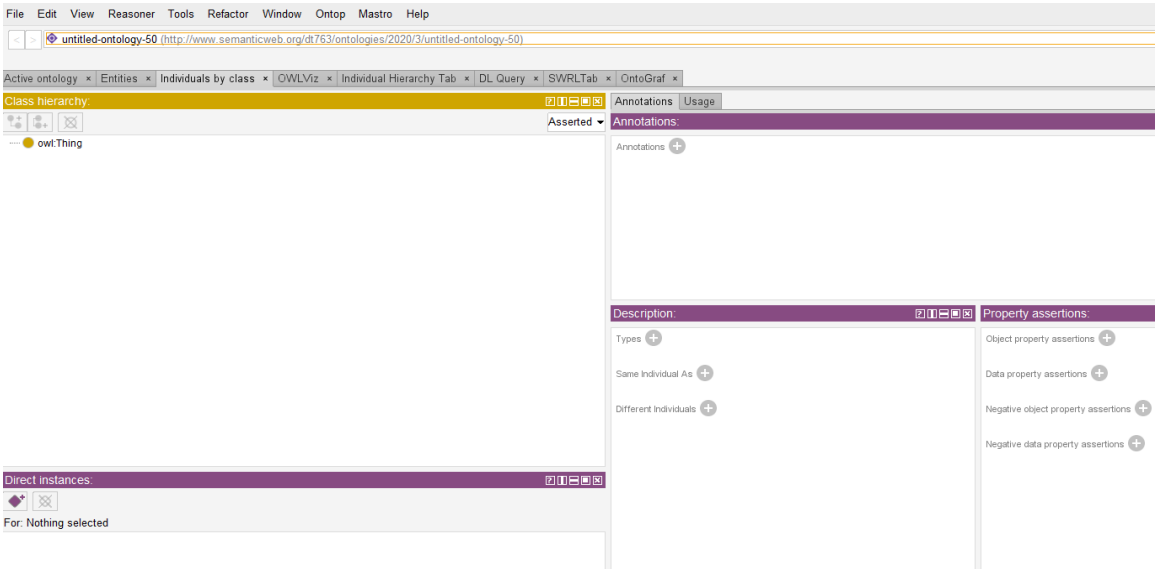

Figure 5: Protege Tool

CHAPTER 4: METHODOLOGY

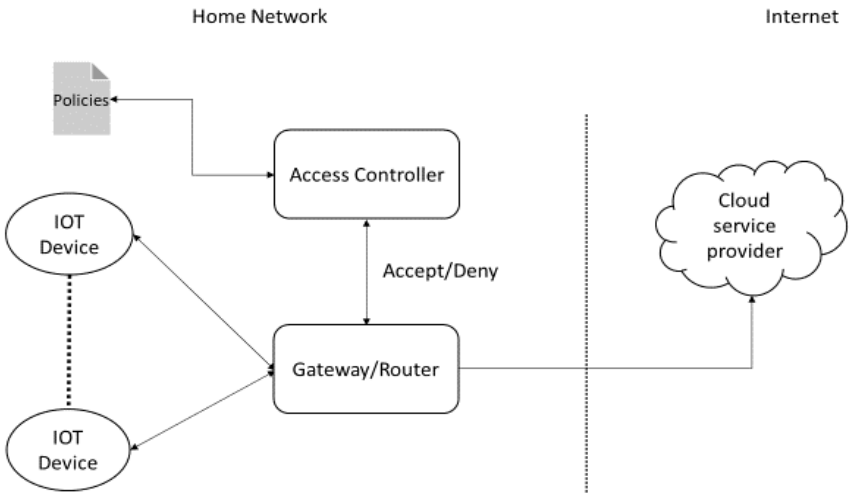

Figure 6: System Architecture

4.1 SYSTEM ARCHITECTURE

The home network consists of a lot of IoT devices like Alexa, smart toaster, smart light, smart plugs, Smart TVs, etc. These devices will connect to the wifi which is even inside the home network through the router. A user uses these IoT devices which collects the data from the interaction or the sensor and sends it across the wifi through the gateway/router to the outside network or the cloud provider.

For example, in our system, Alexa has been configured by the user and the user defines the policy that Alexa should not send data after 11 pm till 5 am. This kind of policy will be

defined for all the IoT devices in the house. Alexa communicates to the outer network or the internet through the network gateway or the router as shown the fig 3 in the system architecture when Alexa tries to communicate to the cloud services or the internet, access controller checks for the policy, and decide to allow or deny the packet. When Alexa try to communicate to the network after 11:30 pm till 5 am, because of some instruction then also the access controller checks the policies defined by the user, it finds that there is a policy which says to deny the packets between 11:30 pm till 5 am then it decides to deny the packet to send to the network outside the home network and protects the device being mutualized.

In the Smart Home Environment, there are a lot of IoT devices that interact with the router or the gateway to fetch or to send the data from the device. The IP address or the packets which the device needs to send across the router need to be monitored and not the packets which are inflow to the device. As we cannot have the IP address access at the router level we can use the external modem and using SPAN [switch port analyzer] we can have a copy of each packet that needs to be sent through the gateway to the internet. We need to disable the DHCP of all IoT devices and enable the static IP address to identify the device. Based on static IP Address and we can look up for the IP address where user policies are defined for the device and we can apply that policy to enable or disable the packet to send to the internet from home network.

4.2 COMPONENTS AND FUNCTIONALITIES

- **Home Network**:

  The network which is set up by the user to control the IoT devices by assigning the Static IP address using SPAN[switched Port Analyzer Network]. The home network is an internal network like LAN wherein the outside world would be unaware of the things going on inside the network. The only way for the IoT devices to connect to the outside world or the internet or cloud would be through the gateway. The home network contains IoT devices in the smart house, users/people residing in the house, router/internet gateway, Access controller which handles the policies defined by the user for the IoT device.

- **Internet**:

  Internet in the diagram refers to the outside network other than the home network where the IoT devices need to Connect through to store the data, interact with the IoT devices through the application or analyze the big amount of sensor data collected from the IoT Devices. The main network the appliances would connect to from the Home network would be almost for the cloud service provider with which the appliance is connected to.

- **IoT Devices/Smart Devices**:

  These are the Smart devices that are in trend and are widely available in the market, for example, google mini, Alexa, smart TV, Smart Lamp, Smart oven, Smart thermostat, Smart Plug, etc. These devices are embedded with one or the other type of sensors they use the instructions which might be default or defined by the user to communicate through the application outside the home network. Smart devices enable devices to be at ease and it enables users to save energy and money. IoT

devices transmit a huge amount of sensitive data which can be personal information, so these devices need to be handled in the right manner with the right access controls for it and also have the right settings, updates on the firmware should not be neglected.

- **Users**:

  Users are the people who reside in the house and they interact with these IoT or Smart devices daily. Users can hold a different role and the role can have different authorization and authentication to the device. The children might not have the same level of access as parents. Similarly, the babysitter who is also the user and will be inside the house at times cannot have the same level of access as Parents. The user we mostly concentrate on here would be the person who has the admin rights or the full control over the network. He/she would be capable of defining the policies which define that when can the device interacts with the internet outside the home network and when it cannot.

- **Internet Gateway**:

  Internet Gateway or the Router is the point where the home network ends and the network outside the home network starts. There are more intelligent gateways available in the market, These are often called Edge Gateways [smart IoT Gateways]. Internet gateway ensures to build a secure bridge between the home network and the internet. These enable trusted connectivity and security which in turn gets the integrity of the system and network together. The importance of the Internet gateway is to have the different Protocols and data bridge which helps to have different data formats. IGW helps in storage to drive intelligent decisions and analysis of edge devices and management. This also helps to have the device

management and to have the access controls in place if policies are defined.

- **Access Controller**:

  This is like an access broker which helps to have access control in place using the policies which are defined by the user for the particular device when it tries to communicate at the wrong time. The device-level access controls can be placed and it helps to have the policies to define the device access to the internet which in turn helps to avoid the attacks. The access control takes up the policies which are defined by the user and evaluates the acceptance or denial for the device using IP through which the device connects and allows the packets to travel accordingly.

- **Policies**:

  Policies are like permissions for the device which are defined by the user according to his/her needs. These Policies are the logical decisions that the user takes and specify some rules or conditions that govern the access of the packets sent by the IoT device to pass the home network. Access policies for the Devices at the router level to avoid unnecessary interaction and help to prevent the intrusion. The right policies at the right place can help the user to mitigate the threat like DDOS attacks which the IoT device is offended for the usage in the recent days. Policies like microwaves should not send any data at night. Depending on the user disabling the toaster to send the data after breakfast time. These might help to add an extra level of security to the IoT Devices. The policies can be written in SWRL which has the same concept as an owl.

4.3 USE CASES

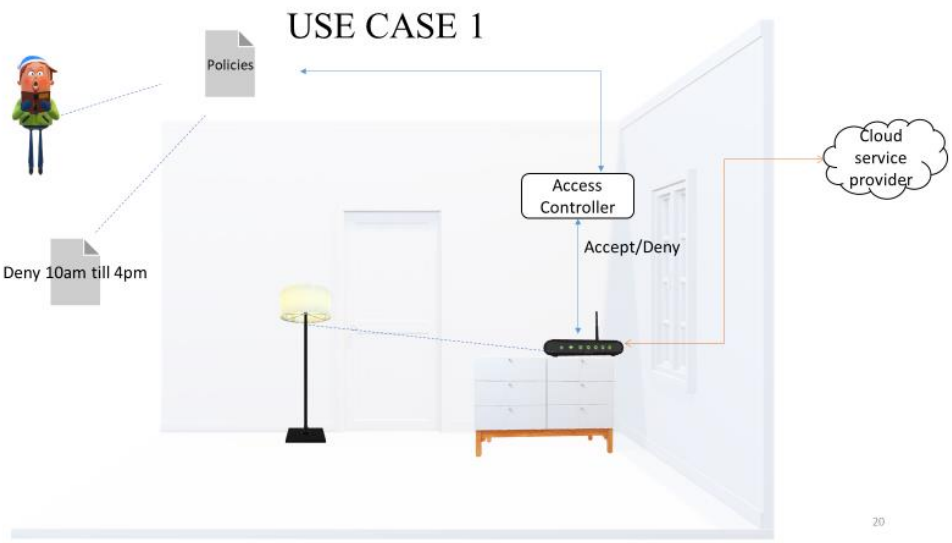

Figure 7: Use Case 1

In this use case, the smart light always communicates through the router, when it tries to communicate to outside the home network, the access controller would fetch the policy defined by the user and evaluates the policy. According to the policy defined the action is taken at a point in time whether to accept or deny packet to traverse further to the network.

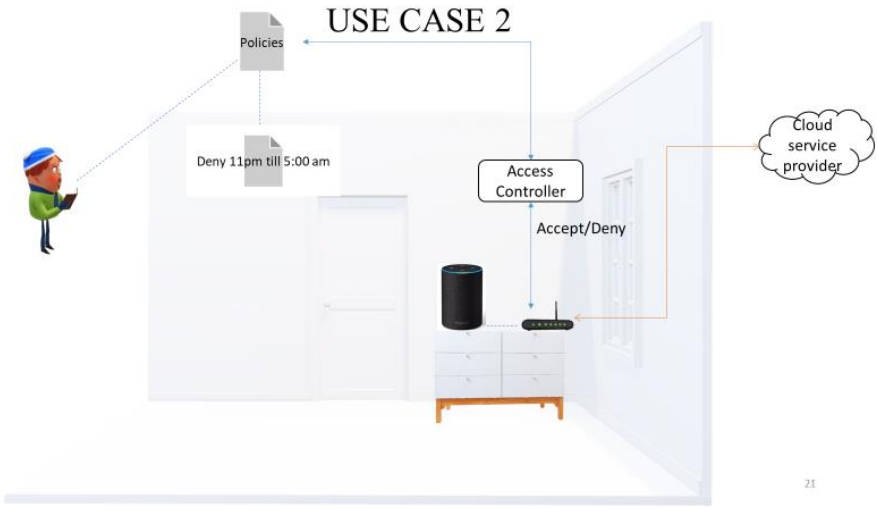

Figure 8: Use Case 2

In this use case the Alexa communicates through the router to cloud service provider when it tries to communicate to outside the home network, the access controller would fetch the policy defined by the user and evaluates the policy. According to the policy defined the action is taken at a point in time whether to accept or deny the packet. There can be several Alexa devices installed in the house which can be used at varied times at different places, these devices should have the access controls defined according to the usage of the people residing inside the house and must be restricted to communicate to the internet at the odd times of the day when they are not in use.

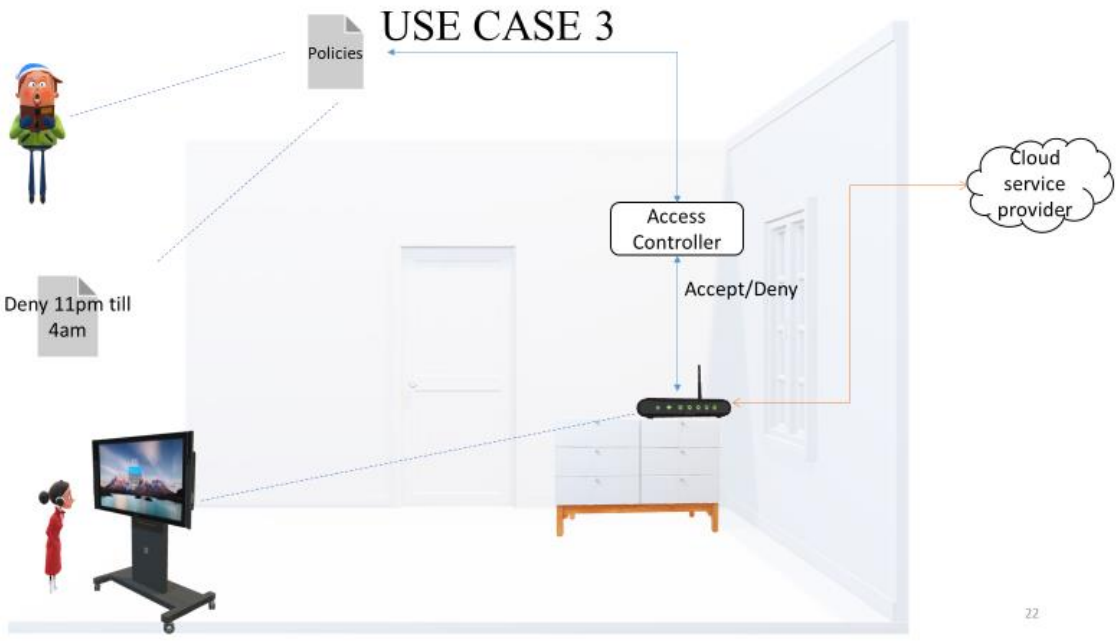

Figure 9: Use Case 3

In this use case the smart device which is used by another person in the house always and it also communicates through the router, when it tries to communicate to outside the home network, the access controller would fetch the policy defined by the user and evaluates the

policy. According to the policy defined the action is taken at a point in time whether to accept or deny the packet.

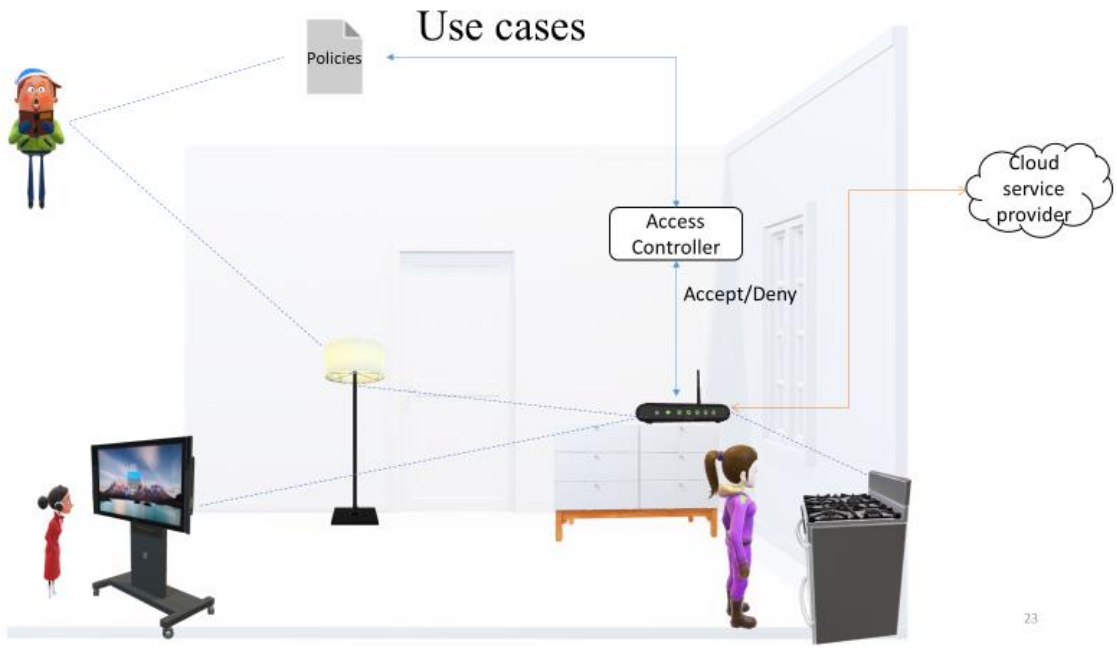

Figure 10: Use Cases

In the conclusion of the use cases, we can understand that different users in the home use smart devices at different points in time. All the smart devices always communicate through the router, when it tries to communicate to outside the home network, the access controller would fetch the policy defined by the user and evaluates the policy. According to the policy defined the action is taken at a point in time whether to accept or deny the packet to traverse further in the network for each device.

CHAPTER 5: IMPLEMENTATION

Figure 11: House Layout

The Implementation is done taking a scenario in the house. Where the house layout is imagined as the above figure.

The house layout is a living room, bedroom, kitchen, dining hall, passage. IoT devices are placed in various places inside the house. These devices are placed in the house layout for the represent IoT devices or the smart devices which will be in the house.

As shown in the image IoT devices are located in the house-

- Alexa in the Living room.
- Smart plug in the kitchen.
- Smart light in the Dining hall.

Smart TV in the Living room. The smart switch or the Smart plug in the kitchen could be connected to the light or any other device like blender or something. Smart Tv and Alexa are placed in the living room. The dining hall has a smart light.

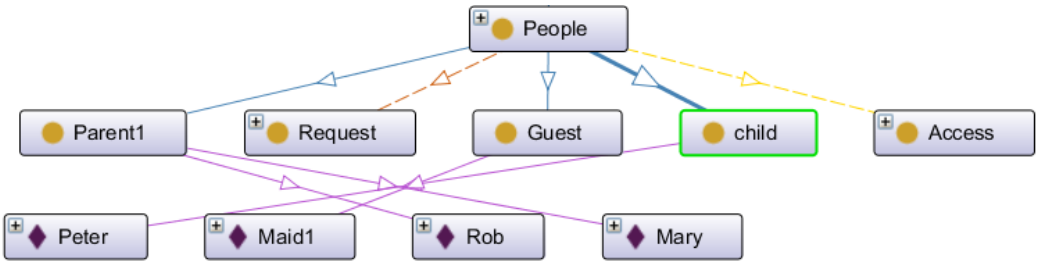

Figure 12: Class: People

The single-family with the parents {husband and wife}, Kid, and a mid/guest is considered to live in the house. Generally, Parents need to have a default or the admin permission for the network and the kid and guest should have limited access to the IoT devices. This is built in a way the small family which has a certain set of smart devices in the home and the rules are built to have better access control and avoid the unnecessary intrusion of the outsiders in the house. The representation of the people in the ontology is shown in the graphical format in figure 12 and the below figure shows how it is being deployed in the ontology.

```
<!-- http://www.semanticweb.org/dt763/ontologies/2020/1/untitled-ontology-12#Mary -->

<owl:NamedIndividual rdf:about="http://www.semanticweb.org/dt763/ontologies/2020/1/untitled-ontology-12#Mary">
    <rdf:type rdf:resource="http://www.semanticweb.org/dt763/ontologies/2020/1/untitled-ontology-12#Parent1"/>
    <rdf:type rdf:resource="http://www.semanticweb.org/dt763/ontologies/2020/1/untitled-ontology-12#People"/>
    <untitled-ontology-12:Initiate rdf:resource="http://www.semanticweb.org/dt763/ontologies/2020/1/untitled-ontology-12#request2"/>
    <untitled-ontology-12:have rdf:resource="http://www.semanticweb.org/dt763/ontologies/2020/1/untitled-ontology-12#Authorized"/>
    <untitled-ontology-12:lives_in rdf:resource="http://www.semanticweb.org/dt763/ontologies/2020/1/untitled-ontology-12#kitchen"/>
</owl:NamedIndividual>
```

Figure 13: Representing people in the knowledge graph

People who reside in the House are Parents, Child and Guest/Maid

- Parents-Rob, Mary who has admin access.
- Child -peter has limited access.
- Maid – Maid1- has no access to the devices.

5.1 ONTOLOGY SCHEMA

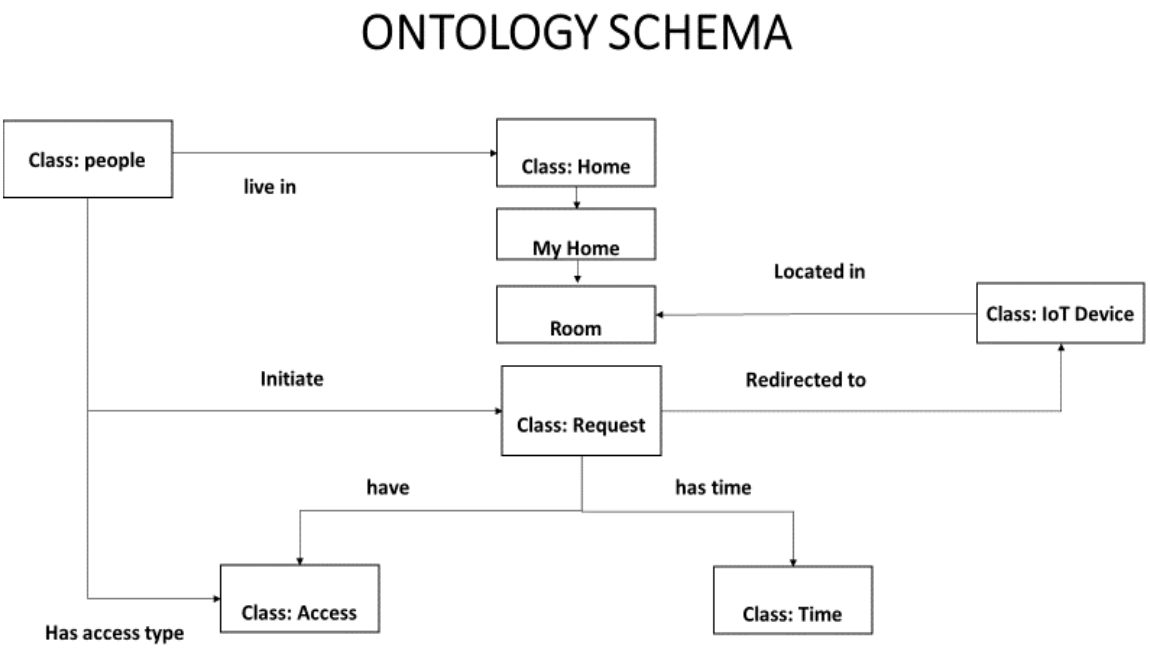

Figure 14: Ontology Schema

As per the schema the ontology has been built using protégé tool. There are classes with the defined labels and the relations are defined on top of the connections which they have been connected. The broad classification of the house environment is done in the above ontology. There are several classes and the connections or relations between them.

In Figure 13: ontology schema We can see

- The people do live in the home which has My home as a subclass and it has a room which would have instances like Livingroom, Kitchen, Dining hall, etc.
- IoT devices are located in room instances like livingroom etc.
- People will initiate the request and do have the access type.
- The Request initiated by the people and that would be redirected to the IoT device which is in the room. The request would be validated by the Access type of the user and the time. After the validation, the request is processed and redirected to the IoT device.

5.2 ACCESS CONTROL

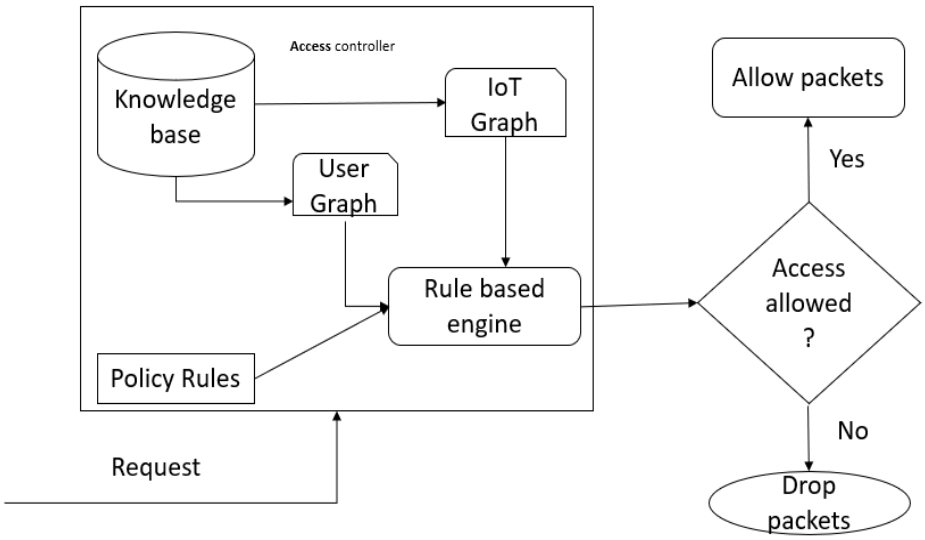

Figure 15: Access Control

The above access controller figure depicts the functionality of the Access controller and the access controller is responsible for the implementation of the access control mechanism of the system. To implement the access control mechanism the access control uses SWRL rules defined in the policies. When the user makes the request to the IoT device and it needs to send the packet out of the home gateway to fetch the data, the IoT devices try to send the packets and it would be diagnosed by the access controller in between. The access controller applies these rules on the user and IoT knowledge graph instance and makes the binary (yes /no) decision. Upon the decision, the packets would be dropped or let it travel further in the network. The packets which are traveling might contain sensitive information like passwords and the username of the application. These packets need to be handled correctly and The SWRL rules or the policies are defined in the upcoming section. The ontology described in the above section classifies the Entities and classes

5.3 ONTOLOGY BUILT-IN PROTÉGÉ WITH USE CASES.

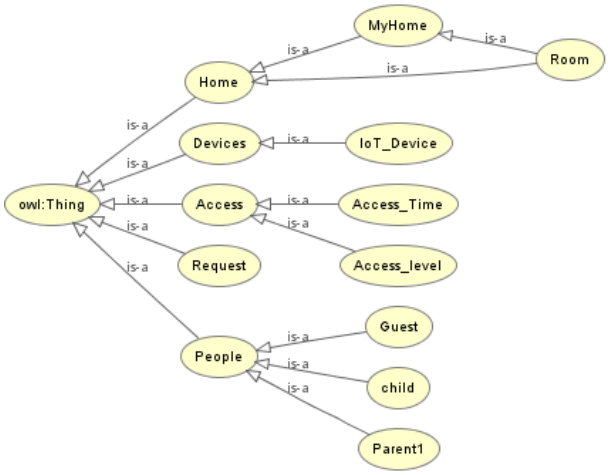

Figure 16: OWL Viz

The otology built in the protégé tool is as shown in the above figure. This the Screenshot of the Protégé tool OWL viz tab. Here we can see the Ontology described in the previous section is deployed. There are 4 classes mainly and the classes are people, Home, Request, Devices, and Access.

```
<!-- http://www.semanticweb.org/dt763/ontologies/2020/3/usingKgforACinSHE#Alexa -->

<owl:NamedIndividual rdf:about="http://www.semanticweb.org/dt763/ontologies/2020/3/usingKgforACinSHE#Alexa">
    <rdf:type rdf:resource="http://www.semanticweb.org/dt763/ontologies/2020/3/usingKgforACinSHE#Devices"/>
    <rdf:type rdf:resource="http://www.semanticweb.org/dt763/ontologies/2020/3/usingKgforACinSHE#IoT_Device"/>
    <located_in rdf:resource="http://www.semanticweb.org/dt763/ontologies/2020/3/usingKgforACinSHE#Living_Room"/>
</owl:NamedIndividual>
```

Figure 17: Representing IoT Device In Knowledge Graph

IoT Devices is having is-a relationship with Devices likewise Access_Time and Access_Level with Access, My home with home, Guest, Parent1, a child with people. These IoT devices are represented in the knowledge graph as shown in the above figure. These relationships are defined based on the classes and subclasses. Each subclass or class will have instances that would have a relation with the other entities in the other classes as defined in the ontology which has been discussed.

Parent1 has instance-Rob, Mary. The child has an instance- Peter, Guest has instance-Maid

Likewise, IoT devices have instance – Smart Light, Smart Plug, Smart Tv, Alexa.

Request has instance -Request1,Request2,Request3,Request4…… etc.

Home has My_Home as subclass which has room in it which has an instance – living, Dining, Bedroom, and kitchen.

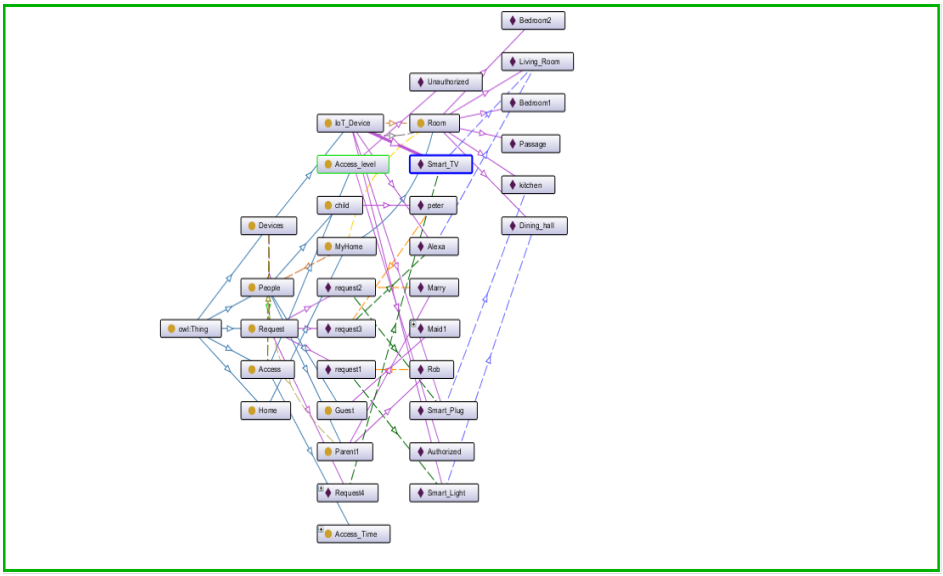

Figure 18: Ontology Built Using Protégé

The whole ontology would look complex wherein we can discuss each entity and the relations along with the rules and policies in this section based on the scenario which this is built on. As mentioned in the above section we have instances of the class or subclass and these instances hold the relationship between the entities of other class which would be as discussed in the ontology section. This would help us to determine and understand the whole system before moving into the scenarios. I have taken examples which would clarify the whole system and who it could be worked and configured. Classes and relations which would imply how the home environment is concerning the smart devices and the people. The requests are processed in the access controller which would be on top of the router to travel in the network further. This decision would be taken by the rules which are specified in the form of SWRL rules in the protégé.

5.4 RULES AND REQUESTS

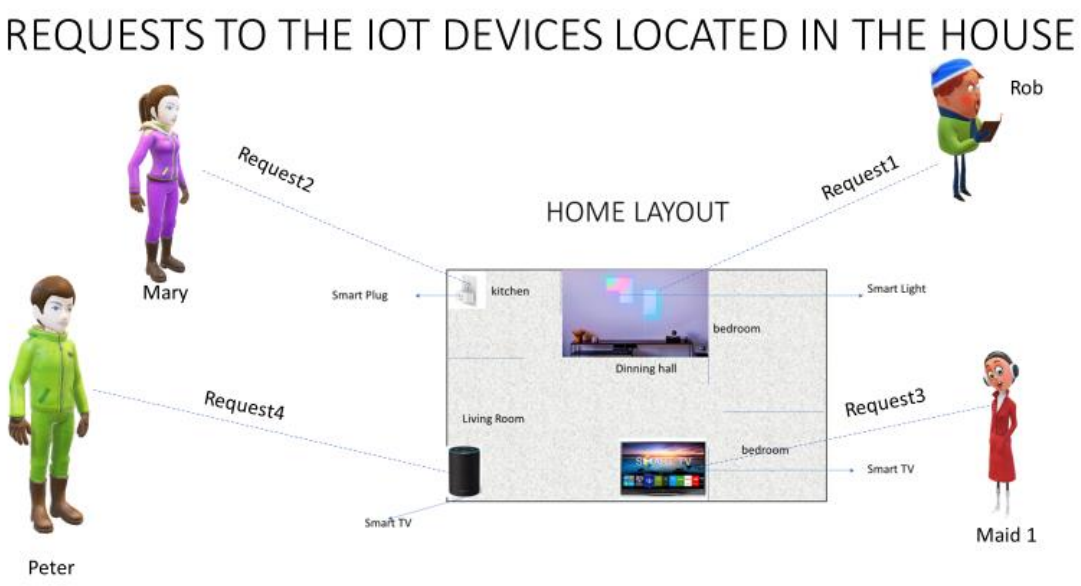

Figure 19: Requests

- ✓ **USE CASE 1- REQUEST 1**
    - Parent1- Rob Initiate Request1 which is for the IoT device- Smart Light located in the dining hall in the house.
- ✓ **USE CASE 2- REQUEST 2**
    - Parent2- Mary initiates Request2 which is for the IoT device -smart plug located in the kitchen.
- ✓ **USE CASE 3- REQUEST3**
    - Child1- Peter initiates Request3 which is for the IoT device -Alexa located in the Living room in the house.
- ✓ **USE CASE 4- REQUEST4**
    - Guest-Maid initiates Request4 which is for the IoT device -Smart TV which is in the Living room in the house.

5.4.1 USE CASE 1- REQUEST1

Parent1- Rob Initiate Request1 which is for the Smart Light located in the dining hall.

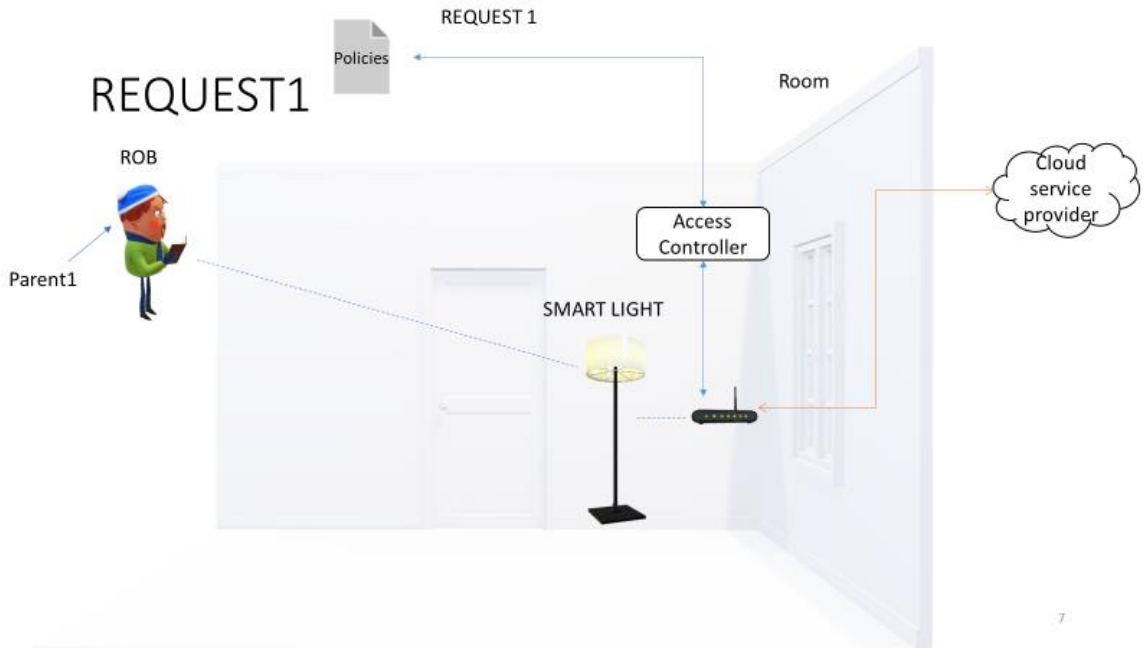

Figure 20: Request 1

Scenario: Parents need to have the default permission or the admin permission which should be given to handling the IoT devices and the rules for them. In this scenario is an example of the parent – rob request should be allowed if he requests any IoT device. In Request1 the parent1- Rob is interacting with the smart light located in the dining hall as shown in the figure. This Request will travel to the router and to the Access controller where it checks the policy, here policies are rules written in the SWRL.

This is implemented in the protégé tool and for the better understanding of the request2, the ontograf view of the protégé is as shown in the below figure.

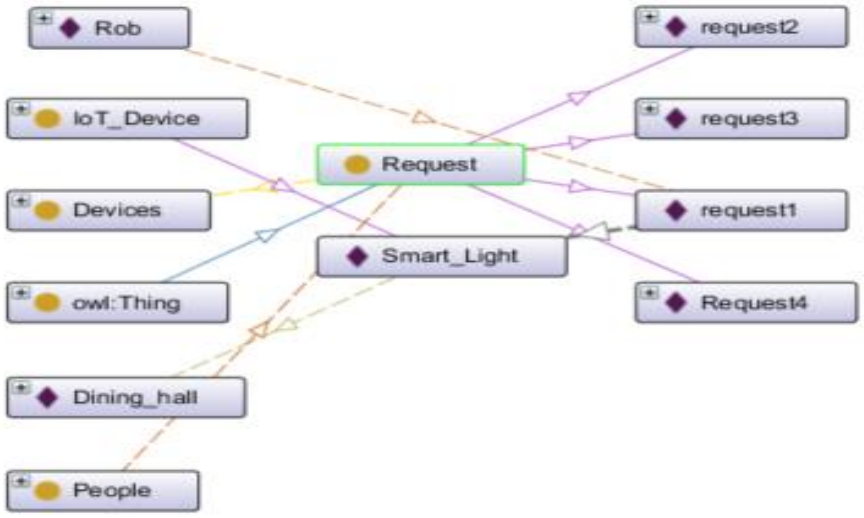

Figure 21: Request 1 In Protege

In this use case, Rob will be having access to all IoT devices at all the location for an unlimited time. The request is an example for the Parent having access to all the IoT devices for which RULE 1 and RULE 2 of the SWRL which are defined would be applied.

Parent 1 [Rob] can access all the devices in the house at all the time

Rob interacts with the IoT device-Smart light and it has to be allowed at any time he requests in smart light which is located in the dining hall in the house.

5.4.2 USE CASE 2- REQUEST2

Parent1- Mary Initiate Request1 which is for the Smart Plug located in the dining hall.

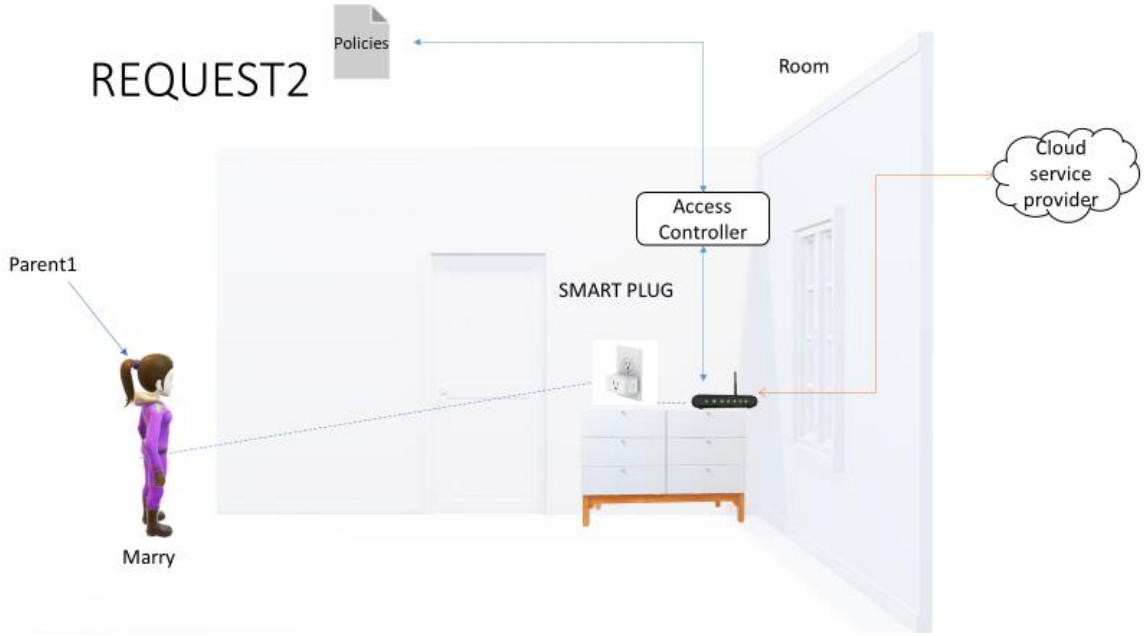

Figure 22: Request 2

Scenario: Parents need to have the default permission or the admin permission which should be given to handling the IoT devices and the rules for them. In this scenario is an example of the parent – Mary's request should be allowed if she requests any IoT device at any point of time in the day. In Request2 the parent1- Mary is interacting with the smart Plug located in the kitchen as shown in the figure. This Request will travel to the router and to the access controller where it checks the policy, here policies are rules written in the SWRL.

This is implemented in the protégé tool and for the better understanding of the request 2, the ontograf view of the protégé is as shown in the below figure.

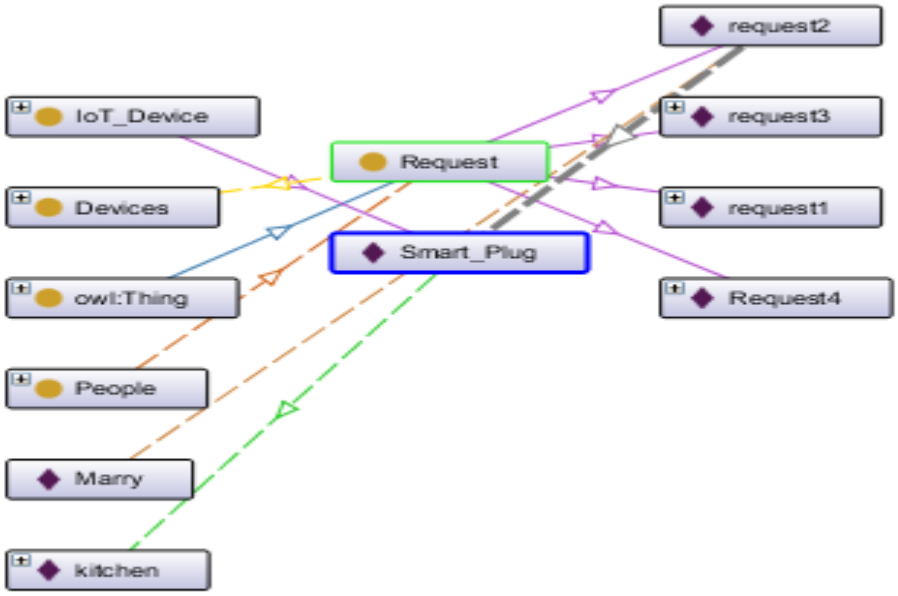

Figure 23: Request 2 In Protege

In this use case, Mary will be having access to all IoT devices at all the location for an unlimited time. The request is an example for the Parent having access to the IoT devices for which RULE 1 and RULE 2 would be applied.

Parent 1 [Mary] can access all the devices in the house at all the time

Mary interacts with the IoT device and it has to be allowed at any time she requests smart Plug which is located in the kitchen in the house. Where this request needs to have the Accept from the policies which have been written as Rule1 and 2 using SWRL.

5.4.3 USE CASE 3- REQUEST3

Child1- Peter initiates Request3 which is for the IoT device -Alexa located in the Living room in the house.

Figure 24: Request 3

Scenario: Children need not have the default permission or the admin permission and the access and permission to handle the IoT devices should be limited. In this scenario is an example of the child – Peter's request to the Alexa should be allowed at a specific time. If he requests any IoT device. In Request2 the parent1- Mary is interacting with the smart Plug located in the kitchen as shown in the figure. This Request will travel to the router and to the Access controller where it checks the policy, here policies are rules written in the SWRL.

This is implemented in the protégé tool and for the better understanding of the request 3, the ontograf view of the protégé is as shown in the below figure.

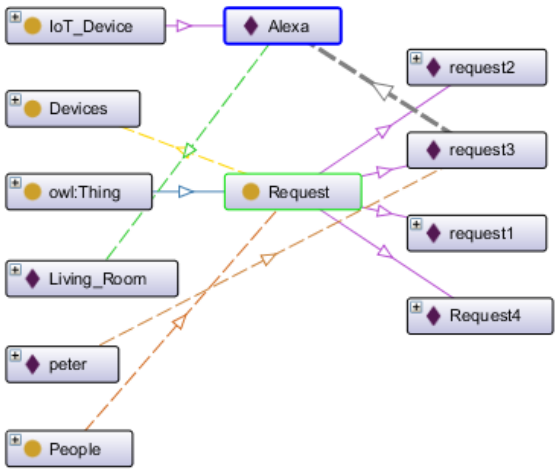

Figure 25: Request 3 in Protege

In this Use case, the IoT devices along with their location and the time and the access are added along with the limited access to the child for the devices.

Child [peter] can access the devices but not all in the house and also in a specific time. This is implemented in the protégé tool and for the better understanding of the request3,

In this use case, Peter will not be having the full access to all IoT devices at all the location for an unlimited time. The request is an example for the child having limited access to the IoT devices. Peter interacts with the IoT device and it must be allowed at a specific time he requests the Alexa which is located in the living room in the house. Where this particular request needs to have the denial from the policies which have been written as Rule3 using SWRL.

5.4.4 USE CASE 4- REQUEST4

Guest-Maid initiates Request4 which is for the IoT device -Smart TV which is in a Living room in the house.

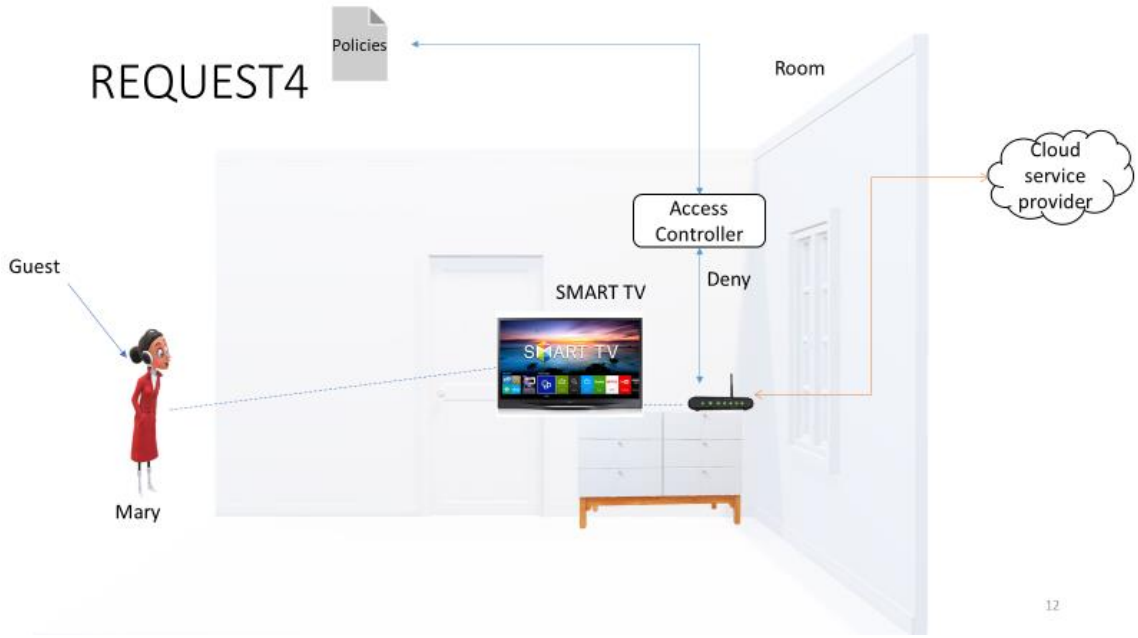

Figure 26: Request 4

Scenario: Maid is treated as the guest in the house, who comes over to the house for a while to complete her work for a while. She represents the temporary people visiting the house. If she tries to access any of the smart devices in the house, she should not be permitted. For example, in the scenario where if the maid1 is trying to operate smart tv present in the living room in the house. The request should be denied. This request is an example and can be applied to all the devices available in the house. At any specific time, any request from the maid should be denied. This sets the example for the case where the request should be denied at all the time and Using SWRL Rule4 can be applied.

This is implemented in the protégé tool and for the better understanding of the request 2, the ontograf view of the protégé is as shown in the below figure.

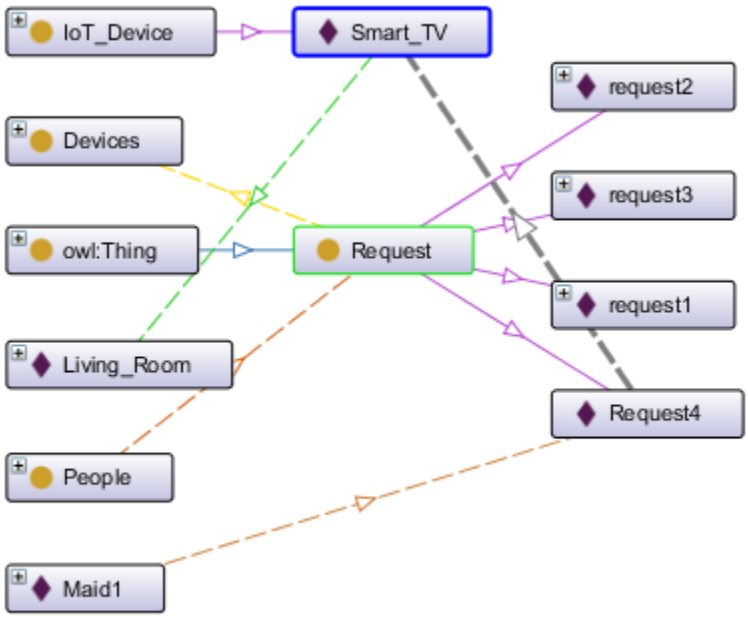

Figure 27: Request 4 In protege

In this use case, Maid1 will not be having any kind of access to all IoT devices at all the location at any given time. This is implemented in the protégé tool and for a better understanding of the request4, Maid1 interacts with the IoT device and it should not be allowed at any given time. She requests the Smart tv which is located in the living room in the house.

Request 4 is an example for the guest not having any kind of access to the IoT devices for which RULE 4 would be applied from the SWRL rules.

Guest [Maid1] cannot access any of the devices in the house and also at any specific time.

5.3 POLICIES

5.3.1 RULE 1 & RULE 2 - PARENT1 DEFAULT ACCESS.

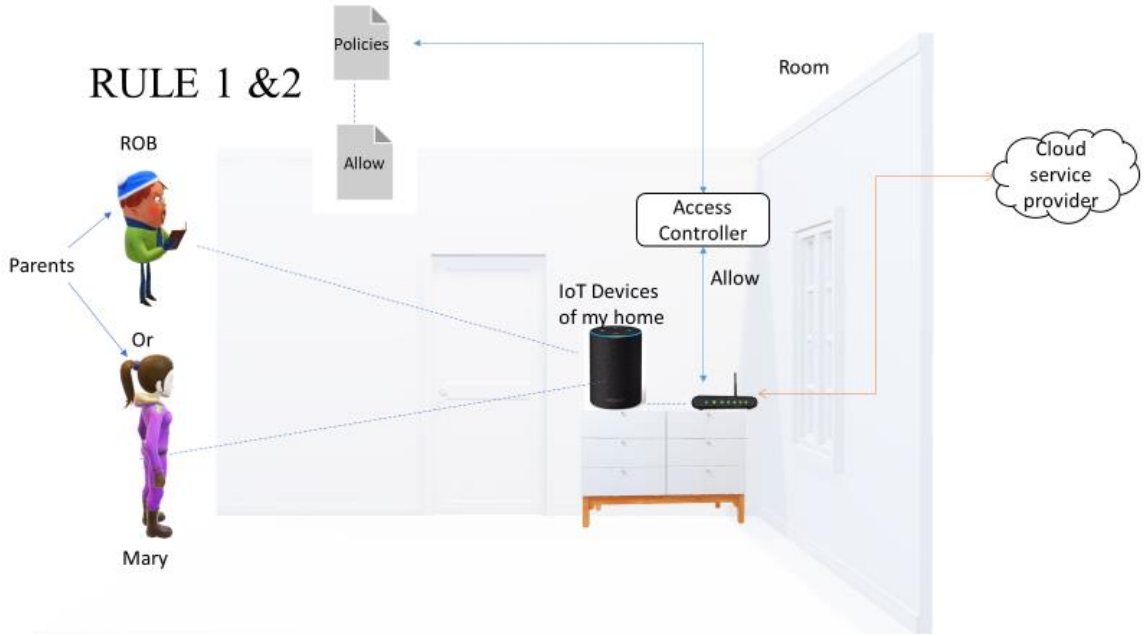

Figure 28: Rule1 & Rule2

Parents are the subclasses of the people. Parents need to have the default permission or the admin permission which should be given. Where Rob is the instance of the parent. Likewise, Mary is the parent and they need to have all the access of the IoT device irrespective of the location and the time. This means they have the authority and the right access to control the rules as well as all the IoT devices in the house. This default permission is given to parents. If the single-family is living with the kids in the house the parents are labeled as the default permission holders.

Rule 1 would be an example for the default permission for the parents and Rule 2 would give the same permission, but it adds location, time, and the access time and permission.

SWRL RULES defined as follows in protégé:

Parent authorized to use IoT devices(default) and request should be accepted when they are made.

RULE 1: In the implementation, the default permission is set for the admin.

Parent 1 [Rob] can access all the devices in the house all the time.

Parent 1 [marry] can access all the devices in the house all the time.

SWRL rules would be as follows:

```
Parent1(?P) ^ IoT_Device(?I) ^Request(?R) ^
Initiate(?P, ?R) ^ Req_time(?R, ?T) ^ directed_to(?R, ?I) ^
swrlb:equal(?t, 1)
-> Allow(?R, true)
```

RULE 2: In the implementation, the default permission over here location, time, access level along with access time are used.

Parent 1 [Rob] can access all the devices in the house at all the time irrespective of location, time, access level along with access time.

Parent 1 [marry] can access all the devices in the house at all the time irrespective of location, time, access level along with access time. SWRL rules would be as follows:

```
Parent1(?p) ^ IoT_Device(?I) ^ Request(?R) ^
Access_level(?A) ^ Access_Time(?AT) ^ Room(?RM) ^
located_in(?I, ?RM) ^ Initiate(?p, ?R) ^ Req_time(?R, ?T) ^
directed_to(?R, ?I) ^
 swrlb:equal(?t, 1)
->Allow(?R, true)
```

5.3.2 RULE 3 CHILD - LIMITED ACCESS.

RULE 3

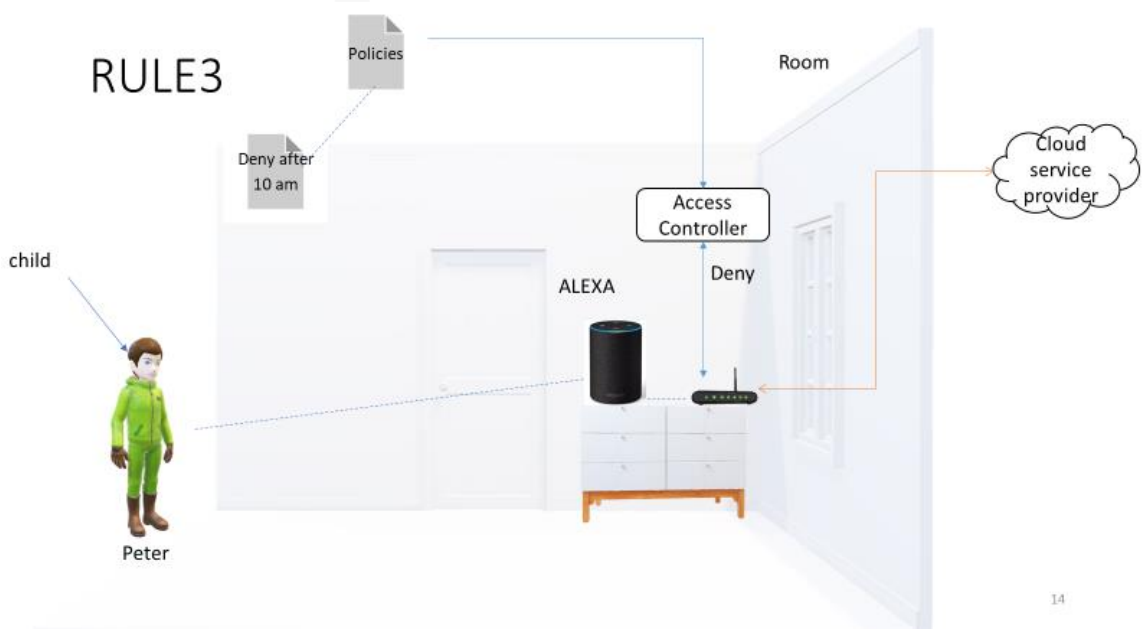

Figure 29: Rule3

The child is the subclass of the people. The child needs to have limited permission and access to the IoT devices located in the house. Here Peter is the instance of the Child, He needs to have limited access to the IoT device irrespective of the location and the time. Which means he should have the authority and the right access to control the rules defined for the particular the IoT devices in the house.

If the single-family is living with the kids in the house with the parents and the permission level will be divided and will be limited to the child. Rule 3 would be an example of the limited permission for the child and would give the same permission concerning location, time, and the access time and permission. Here if a child tries to interact with the Alexa the packets should be dropped.

RULE 3: In the implementation, the limited permission will be implemented this adds location, time, access level along access time to the rules.

Parent 1 [child] can not access Alexa in the house concerning the specified location, time, access level along with access time.

SWRL rules would be as follows:

```
child(?c) ^ IoT_Device(Alexa) ^ Request(?R) ^
Access_level(Authorized) ^ Access_Time(?AT) ^ Room(?RM) ^
located_in(Alexa, ?RM) ^ Initiate(?c, ?R) ^ Req_time(?R, ?T)
^ directed_to(?R, Alexa) ^
swrlb:equal(?t, 1)
-> Allow(?R, false)
```

5.3.3 RULE 4 GUEST - NO ACCESS.

RULE 4

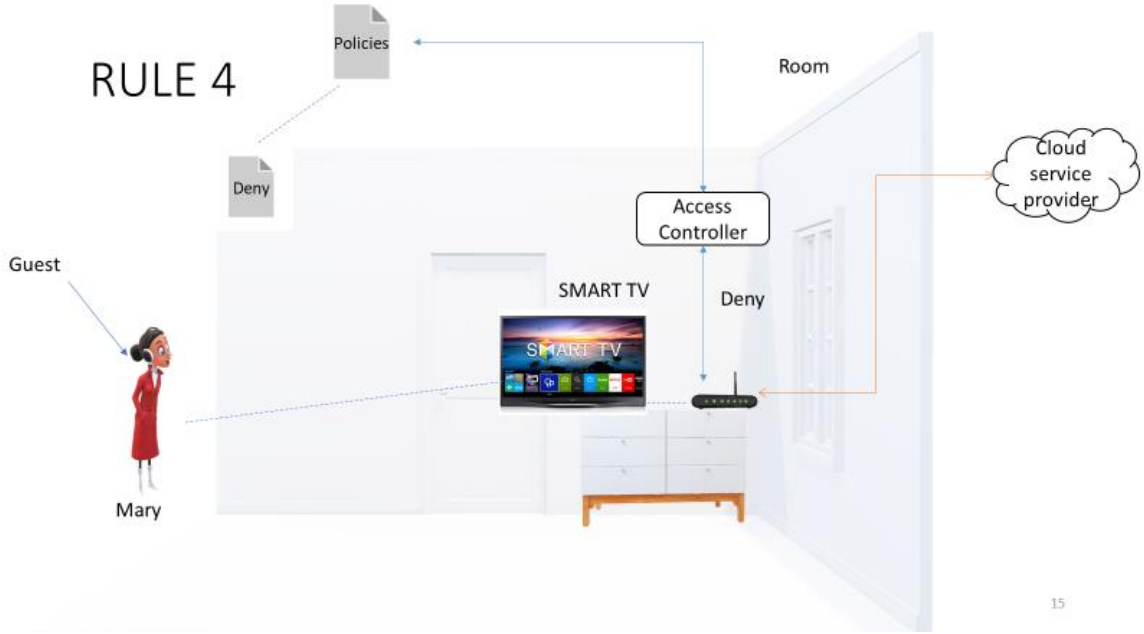

Figure 30: Rule4

Guest is the subclasses of the people. Guests need not have any kind of permission and access to the IoT devices located in the house. Here Maid1 is the instance of the guest, She needs to not have any kind of access to the IoT device irrespective of the location and the time. Which means she should not have the authority and the access to control the rules defined for the IoT devices in the house.

If the single-family is living with the kids in the house with the parents and the permission level will be divided and no permission should be granted to the guest. Rule 4 would be an example of not having any kind of permission for the guest and would not have permission with irrespective of location, time, and the access time and permission. Here if guest- maid1 tries to interact with the smart tv the packets should be dropped.

RULE 4: In the implementation, any kind of permission will not be given would be implemented this adds location, time, access level along access time to the rules.

Guest [Maid1] cannot access the Smart TV of the house in the specified location, time, access level along with access time.

SWRL rules would be as follows:

RULE 4

GUEST NOT AUTHORIZED

```
Guest(?g) ^ IoT_Device(Smart_TV) ^ Request(?R) ^
Access_level(?A) ^ Access_Time(?AT) ^ Room(?RM) ^
located_in(?I, ?RM) ^ Initiate(?g, ?R) ^ Req_time(?R, ?T) ^
directed_to(?R, ?I)
^ swrlb:equal(?t, 1) -> Allow(?R, false)
```

CHAPTER 6: EVALUATION

6.1 SWRL RULES EVALUATION AND EXECUTION

- SWRL RULE1 AND RULE2

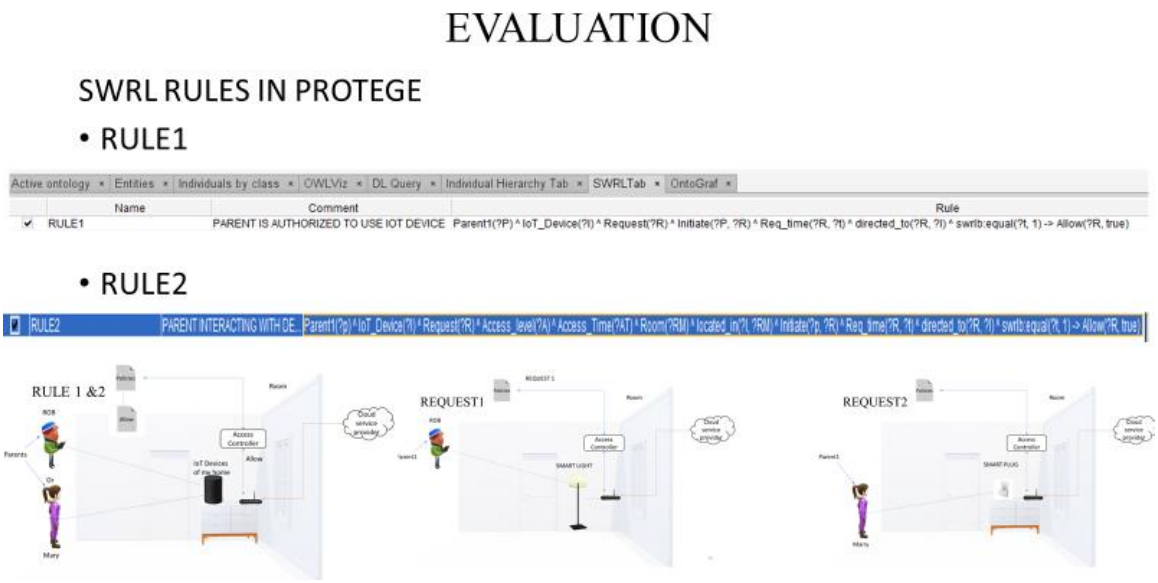

Figure 31: Evaluation - Rule 1 & Rule 2

The rule 1 and rule 2 of SWRL are the generic permission for the parents wherein it says the default permission for parent is given whenever he/she interacts with the IoT device will be allowed. This means the Request1 and Request2 which are from parents are allowed at every point of time.

The execution of these SWRL rule 1 and SWRL rule 2 can be seen in figure 32.

This rule execution is done in 3 steps.

- transfer SWRL rules and relevant OWL knowledge to the rule engine.

- Run the rule engine.
- Transfer the inferred rule engine knowledge to owl knowledge

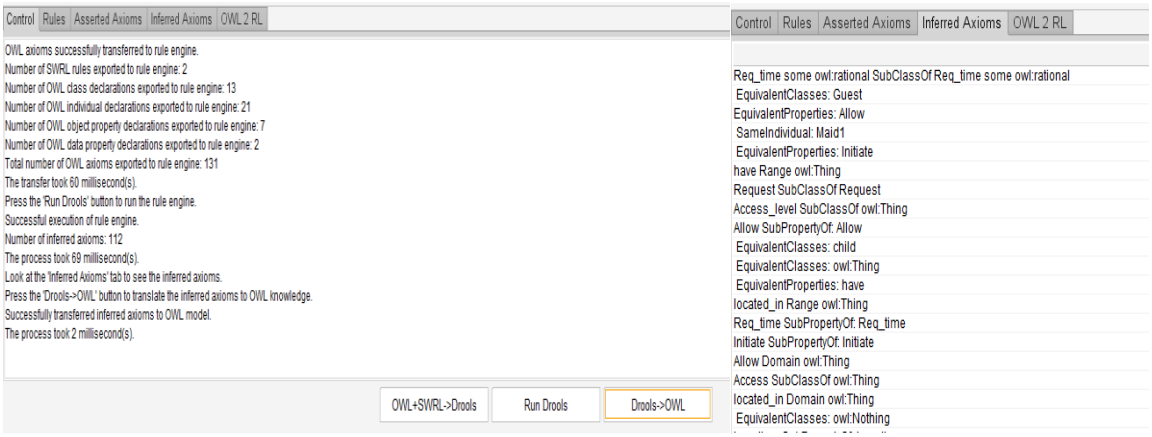

Figure 32: Execution - SWRL Rule 1 & Rule 2

We would be transferring SWRL rules and relevant OWL knowledge to the rule engine by using the 'OWL+SWRL->Drools' button. We have to run the rule engine using the 'Run Drools' button which gives the inferred axioms. These inferred rule knowledge will be converted to owl knowledge by using the 'Drools->OWL'.

Over here we can see that the number of rules executed is 2 and they are converted to the inferred axioms which would be like in the right side where the inferred axioms tab is displayed.

➢ SWRL RULE 3

Rule 3 of SWRL is the limited permission for the child wherein it says the default permission for a child should not be there and as per the request 3 scenarios, the rule is written is given whenever child interacts with the IoT device request will be allowed

as per the request time. If the request time is not in the specific time for example if the time of the request is after 10om then the request will be denied. This means the Request3 which is from a child will be allowed at every specific time quoted. These rules are quoted in protégé as shown in the below figure.

EVALUATION- SWRL RULE3

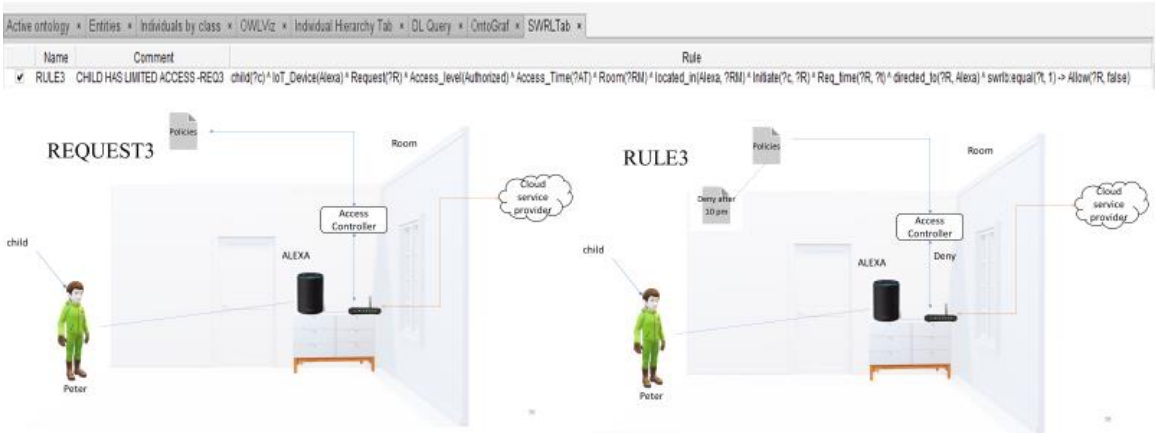

Figure 33: Evaluation - SWRL Rule3

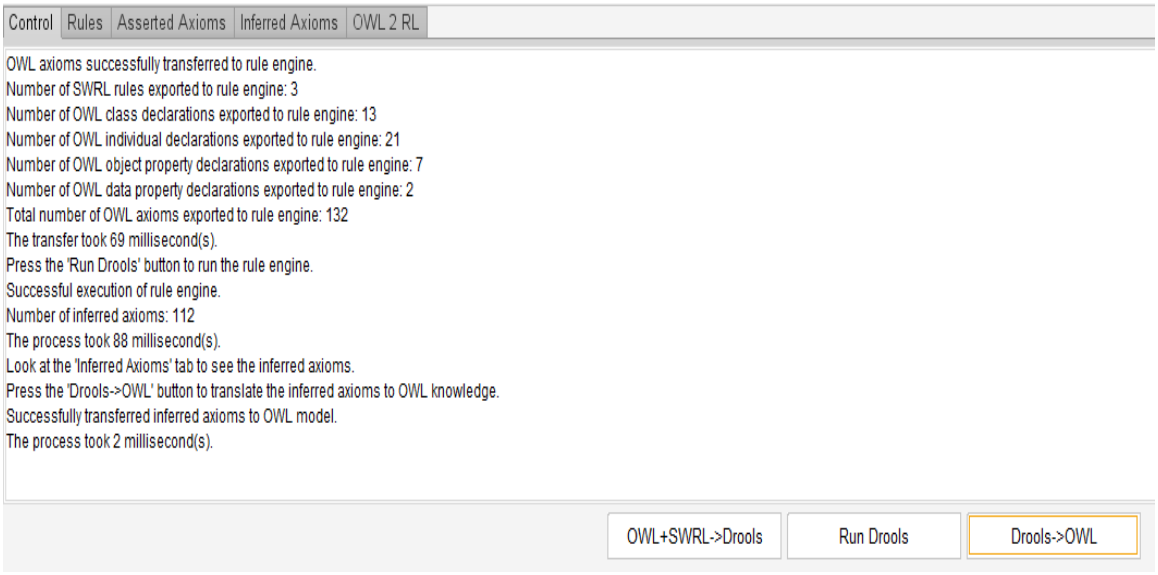

Figure 34: Execution - SWRL Rule 3

As shown in figure 34 The SWRL rule 3 will also be executed similarly to rule 1 & 2. This rule execution is done in 3 steps.

- transfer SWRL rules and relevant OWL knowledge to the rule engine.
- Run the rule engine.
- Transfer the inferred rule engine knowledge to owl knowledge

➢ SWRL RULE 4

EVALUATION-SWRL RULE 4

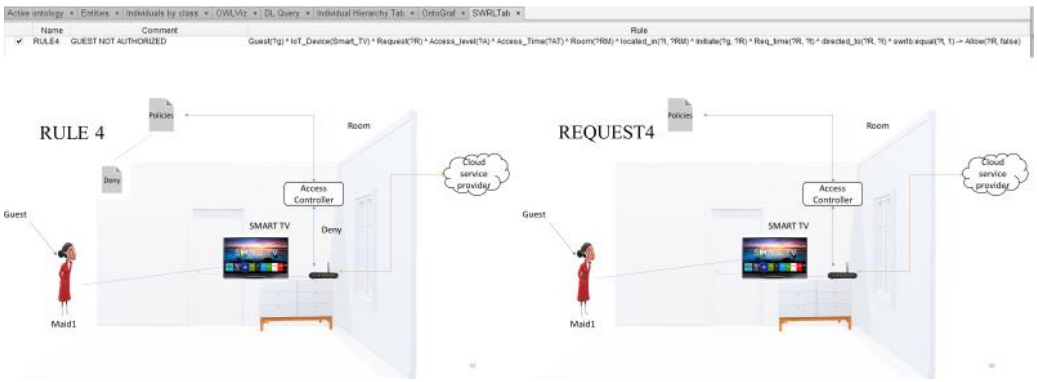

Figure 35: Evaluation - SWRL Rule 4

Rule 4 of SWRL suggests that for the guest-maid no permission should be there and as per the request 4 scenarios, the rule is written is given whenever maid interacts with the IoT device – smart tv request will be denied at any point of time. This means the Request4 which is from maid will be not be allowed at any specific time. These rules are quoted in protégé as shown in figure 35.

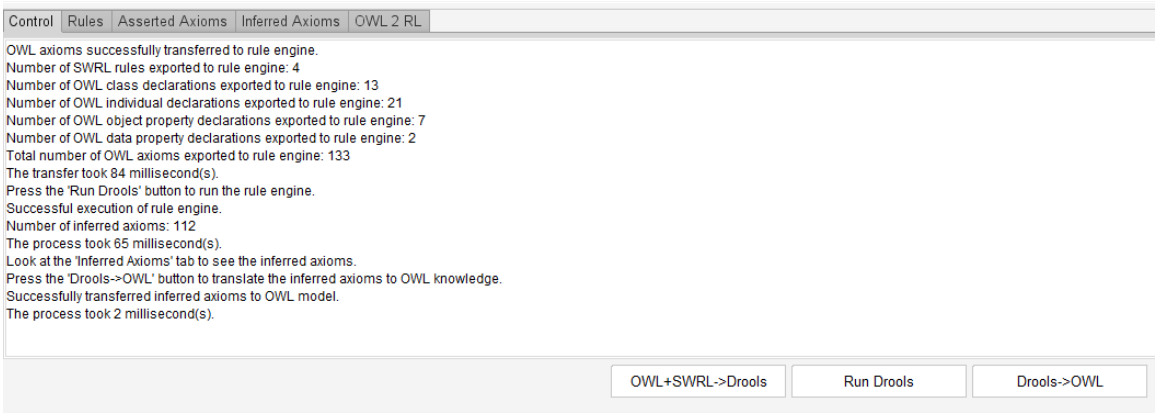

Figure 36: Execution - SWRL Rule 4

As we can see in figure 36 all the SWRL rules mentioned are executed successfully and this is done in steps. We would be transferring SWRL rules and relevant OWL knowledge to the rule engine by using the 'OWL+SWRL->Drools' button. We have to run the rule engine using the 'Run Drools' button which gives the inferred axioms. These inferred rule knowledge will be converted to owl knowledge by using the 'Drools->OWL'.

To evaluate the system, we have created the generic use cases and the situation which could happen in real-time. This implementation of the rules is based on generic scenarios that are explained in the requests in the Use case section. The defined rules are in the policies section would apply to many scenarios where it can be used other than mentioned. The rules would run on the Rule-based engine as described in the access controller section. After the rules are applied the access controller would deny or allow the request made. In real-time, this concept can be used, and the packets can be captured which are flowing through the router and it can be dropped using the rules defined in the policies section.

CHAPTER 7: BUSINESS IMPACT

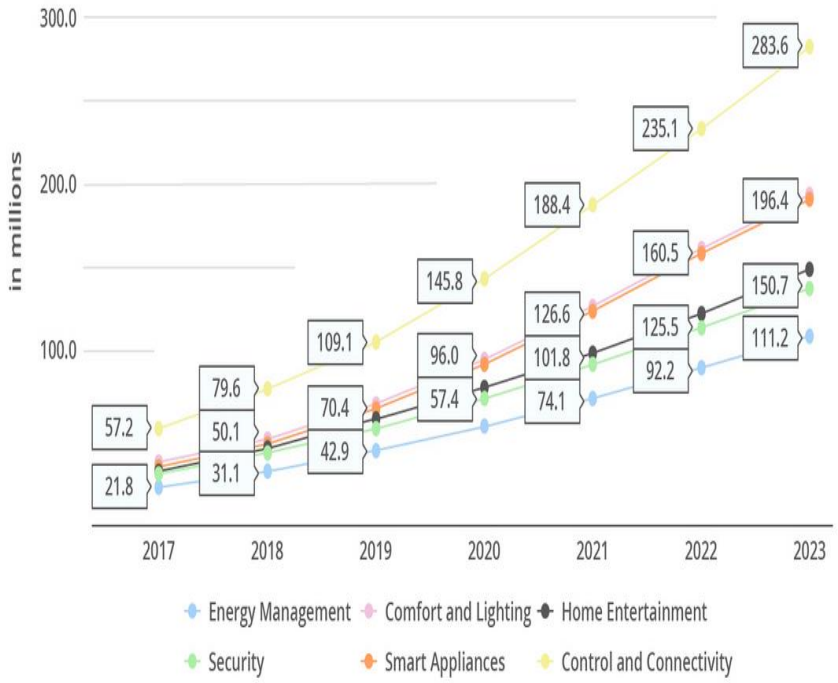

Figure 37: Smart Home Technology Growth

The above Graph shows the spike in the improvement of the Smart home environment section from 2017 to 2023. The Smart appliances purchase is 21.8 million which is expected to reach 196.4 million in 2023 which in turn says the increase in the controls and connectivity from 57.2 million in 2017 and predicted to reach 283.6 million by 2023. These numbers indicate the huge progress in the field and the need of having the right security and controls in place so that the customer is not prone to the cyber-attacks which has been majorly indicated and started its hype now only. Having the right access controls in place with the security of a smart home environment is quite essential.

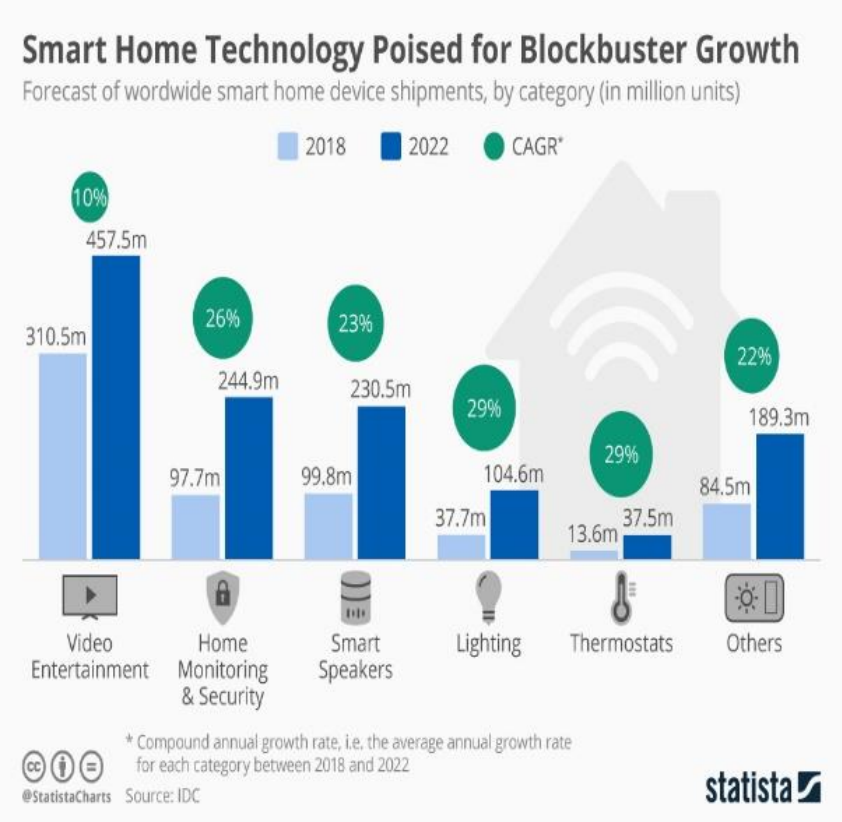

Figure 38: Statista -Smart Home Technology Growth

From the above graph, we can see that the categories of smart home appliances' growth. This even indicates the major increment in the smart home monitoring and security section which is having growth from 97.7 million in 2018 and predicted to reach 244.9 million which is 26% of the growth by 2023. Which explains the necessity of concentration in this region is required. The other section like thermostats lighting is predicted to have a growth of 29% by 2020. Though home monitoring and security section stand second it holds more importance than the thermostat and lighting as home monitoring and security are important to protect all the IoT devices.

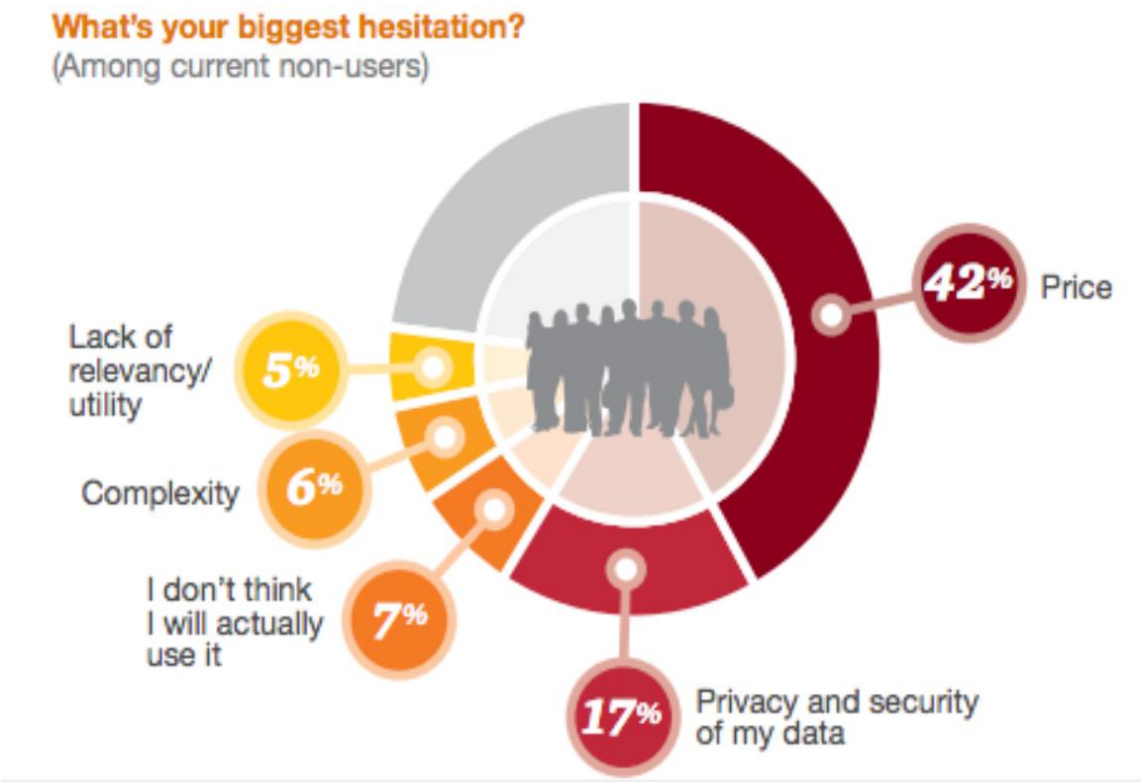

Figure 39: PWC Survey result

50 billion IoT devices will be connected by 2020 is the estimation of the Gartner. According to a new study from PwC, consumers need to have the confidence to adapt to the idea of a smart home environment. From their survey, it is found that though 81% of the participants were aware of the concept of a smart home, only 26% own smart home. When the survey was conducted to know what their biggest hesitation is to adapt to the smart home we could see from the chart that 42% of people believe that it is costly and the price is their major concern, The second biggest concern among people which could be changed is 17% privacy and security concerns. The righteous practice is essential to define for the way people should handle the IoT devices or the smart appliances inside the smart home. Access controls for the devices could be the first step towards it.

CHAPTER 8: CONCLUSION AND FUTURE WORK

8.1 CONCLUSION

This project enables to have the right access controls in place by using the policies as per the user and defined by the user. An introduction outlines the project idea that in this Project using knowledge graph the access controls for the smart home devices. The packets sent by the IoT devices are scrutinized in the gateway level through access controller which fetch the latest policies defined by the user and it decides to allow the packet or deny the packet according to the rule set by the user.

The literature review gave insights into many relevant papers. The semantics of the web technology has been stated and started as early as in the 20th century but came into extensive usage and trend after 2012 when Google declared that they would be using a knowledge graph to connect the data which would help to search in the google. Many papers give insight into the smart home environment and the interaction between the smart device and the gateway, the importance of having access controls in place, graph of things, OWL language usage, types of data from the IoT devices, The Knowledge graph definition.

In the Implementation, I have outlined the system architecture which states smart appliances inside the house which send data to the cloud provider through the internet gateway. These are converted to static IP addresses and sent to an access controller to lookup for the policies defined by the user and decide whether it has to allow or deny in real-time. According to the policies defined in SWRL and the requests mentioned will be evaluated and executed in the protégé as suggested in the evaluation.

8.2 FUTURE WORK

In the future, I would like to extend the same concept being applied in offices. I want to explore how this could be applied over there as it might have a different set of people and the access control which would be taken into consideration while dealing with the IoT or smart devices inside the office buildings or commercial buildings etc. I would also like to explore more on smart building or office setups. I would like to even implement the rules and the policies in the more complex setup of the house would be working parallel with the access controller to have the right decision to be made. Implementation of the right access control in the other direction other than the home environment would be on the next level which would be the connected smart apartments. I would like to explore more about the ontology tools and its usage like the Gene ontology tool which can be used for the complicated and the large ontologies.

GENE ONTOLOGY (GO) TOOLS

This is an ontology tool that is developed by developed within the Bioinformatics Group at the Lewis-Sigler Institute. This tool can handle more complicated ontologies. This tool can be explored to make further studies on the complicated ontologies which like smart buildings and cities or the connected and complicated ontologies. This tool can be explored to build more complex ontologies.

Other tools in the market can also be used to define the ontologies like NeOn, Swoop, semantic turkey, Knoodl, Anzo for excel, etc.

References

[1] Paolo Barsocchi, Antonello Calabrò, Erina Ferro, Claudio Gennaro and Eda Marchetti and Claudio Vairo, "Boosting a Low-Cost Smart Home Environment with," 2018.

[2] Vijay Sivaraman, Hassan Habibi Gharakheili, Arun Vishwanath, Roksana Boreli, Olivier Mehani, "Network-Level Security and Privacy Control for Smart-Home IoT Devices".

[3] Tam Le, Matt W. Mutka, "Access Control with Delegation for Smart Home Applications," Michigan State University.

[4] Ji Eun Kim, George Boulos, John Yackovich, Tassilo Barth, Christian Beckel, Daniel Mosse, "Seamless Integration of Heterogeneous Devices and Access Control in Smart Homes," 2019.

[5] Mohammed Alshahrani, Issa Traore, "Secure mutual authentication and automated access control for IoT smart home using cumulative Keyed-hash chain," 2019.

[6] Georgios Mantas, Dimitrios Lymberopoulos, Nikos Komninos , Chapter in Wireless Technologies for Ambient Assisted Living and Health Care: Systems and Applications, chapter 10 Security in Smart Home Environment.

[7] Weijia He, Maximilian Golla, Roshni Padhi, Jordan Ofek, Markus D¨urmuth, Earlence Fernandes, Blase Ur, "Rethinking Access Control and Authentication for the Home Internet of Things (IoT)," 2018.

[8] Sowmya Ramapatruni, Sandeep Nair Narayanan, Sudip Mittal, Anupam Joshi, and Karuna Joshi, "Anomaly Detection Models for Smart Home Security".

[9] Heiko Paulheim, Philipp Cimiano, "Knowledge Graph Refinement: A Survey of Approaches and Evaluation," *IOS Press,* 2016.

[10] Maithilee Joshi, Sudip Mittal, Karuna P. Joshi and Tim Finin, "Semantically Rich, Oblivious Access Control Using ABAC for Secure Cloud Storage".

[11] Narendra Anand, Srinivas Yelisetty, Anuraag Chintalapally , Colin Puri , Teresa Tung, Michael Giba, "A Knowledge Graph Driven Approach for Edge Analytics",industrial internet consortium," 2017.

[12] Cheng Xie, Di Liu, Yun Yang, Po Yangz, Beibei Yu, Zhibo Cheny, Qiang Fengx, Jiqin Peng, "Knowledge Graph based Internet of Things Middleware," 2019.

[13] Danh Le-Phuoca, Hoan Nguyen Mau Quoca, Hung Ngo Quoca, Tuan Tran Nhata, Manfred Hauswirthb, "The Graph of Things: A Step Towards The Live Knowledge Graph of Connected Things".

[14] D. L. McGuinness, F. Van Harmelen et al., "Owl web ontology language overview, W3C recommendation, vol. 10, no. 10," 2010.

[15] N. K. Sharma and A. Joshi, "Representing attribute based access control policies in owl," 2016.

[16] K. Lalana, "A policy language for the me-centric project," TechReport, HP Labs, 2002.

[17] Maria Bermudez-Edo,Tarek Elsaleh,Payam Barnaghi,Kerry Taylor, [Online]. Available: IoT ontology https://www.w3.org/Submission/2015/SUBM-iot-lite-20151126/.

[18] Ian Horrocks, Peter F. Patel-Schneider, Harold Boley, Said Tabet, Benjamin Grosof, Mike Dean, " SWRL: A Semantic Web Rule Language," [Online]. Available: https://www.w3.org/Submission/SWRL/.

[19] Daniel Elenius, Susanne Riehemann, "SWRL-IQ User Manua," 2012.

[20] "Protege 5 Documentation," [Online]. Available: http://protegeproject.github.io/protege/getting-started/.