CHAPTER 7
RBSEE Architecture: The Proposed Approach for Handling IoT Big Data

7.1 Introduction

This chapter deals with those big data which are generated from IoT devices and systems. Since the inception of the Internet of Things (IoT) technology, the range and domain of datasets in many cases has expanded in multiple dimensions [40]. Along with such expansions, IoT system involves synchronization, collaboration and extension of thousands of sensors, actuators and transponders. To convert the data generated from such varied devices into a standard format possesses baffling challenges. Furthermore, with the amount of information generated and transmitted from these devices (which is largely unstructured) it becomes really hard for the system to be instantaneously responsive with minimum possible latencies. Today, IoT technology is slowly but surely penetrating in every aspect of daily life. No doubt, IoT has great potential to provide modern consumer with convenience but at the same time due to its restrictive and constrained nature its potential vulnerabilities increases. Therefore, there is an urgent need to address these vulnerabilities in order to fully exploit the potentials of IoT technology in a secured and privacy-preserved manner ([13], [20-21], [71], [88], [103-104], [114], [129], [149], [158], [174], [185-186], [190-191], [199]). In recent times we have seen an increasing number of a variety of tools and systems for the purpose of effectively handling big data. Among them, Hadoop has emerged as one of the most prominent and favoured framework. Although Hadoop outshines all other frameworks available for handling large volumes of data, however, it has severe limitations while handling real-time requests. In any typical IoT ecosystem, real-time and instantaneous handling of request and dataset is pivotal and a fundamental property. This chapter aims to propose energy efficient distributed system architecture for persuasive, privacy-preserved and real time handling of data generated from IoT devices. This architecture is called by the name “Request-Based, Secured and Energy Efficient (RBSEE)” architecture [8]. The current chapter is divided into five sections. The second section talks about the preliminaries about the concept along with the need of securing IoT systems and devices. The third section gives
an introduction about the wireless sensor networks (WSN). The fourth section provides a detailed overview of the proposed architecture. The various components and their corresponding working have also been explained in this chapter. The final section presents the conclusion wherein the advantages, uses and applications of the proposal are mentioned.

7.2 Need for Securing IoT Devices

The constrained and limited memory, low power consumption and limited processing capabilities are some of the restrictions that are attached to any typical sensor network. If the IoT devices are left unsecured there are chances that the hackers can cause adverse effects which may directly or indirectly affect the user or consumers. Let us take few examples to understand the need of securing IoT devices and system.

Example 1:

Suppose that we have a person having an IoT enabled pace maker embedded in the body. If a hacker hacks the devices or the system and stops the pace maker then in this situation what will happen to the person? This can cause severe effects and even cause the death of the person!

Example 2:

A Person is having an IoT enabled Smart home automation system. He left for work leaving the kids alone at home. Suppose that some hacker hacks into the system and opens the gate and get hold of the kids. In this situation what can happen? The kids can be kidnapped by the intruders apart from burglary.

Example 3:

Suppose the home is equipped with cameras. You are spending some leisure time with the family members, and suppose that someone hacks into the system and uploads the video on the internet. This can cause adverse after-effects and can compromise the privacy and security of the individuals.

Thus securing the IoT device is highly crucial in order to protect the privacy of the users and their data and information.

7.2.1 Security Issues in IoT devices
Although the security of IoT devices is of utmost importance, yet there are several hindrances in the path of securing these devices. Some of these are briefly explained below.

7.2.1.1 *Inability to understand the potential risks involved in adopting IoT technology*

Most of the manufacturers are not well aware of the potential threats and vulnerabilities associated with adopting IoT technology for their business. Most of the time they rely on third-party solutions to overcome these issues. Primarily there are two kinds of risks involved. These are mentioned below.

(i) *Risks from the manufacturer’s point of view*

For the sake of business and increasing the profits and market shares, most of the manufacturers are blindly developing IoT devices without implementing proper security mechanisms. Their primary aim is to capture the market for selling the product as fast as possible. The security issues generally neglected or take a back seat. These IoT devices sometimes collect highly sensitive and private data without the knowledge of the consumers which must not be exposed to the outside world. The manufacturer of an IoT device which measures the temperature, humidity and pressure of the environment often argue “what is the need for implementing security features in such IoT devices” without understanding that these data in combination with other data from other sensors can provide several insights into the user’s routine and tastes which can be used by the promotional companies for user profiling.

(ii) *Risks from the consumer’s point of view*

The consumers don’t want to waste their time to configure the security modules of the IoT devices and generally opt for plug and play devices with minimal or no external interventions. It is often observed that even the passwords of these devices are left to default values and never changed by majority of the users. This gives the hackers a cakewalk in hacking the devices as most of the default password are in public domain which can be easily cracked. Most of the times the consumers don’t care about what information are being captured by the sensors of the IoT devices, rather they are interested only in the service they have asked for. We have often observed that whatever we talk about on social media, the advertisements of same things start pouring in our inboxes. All this happens because of the information captured by these social networking websites are shared with the promotional
companies which perform the profiling of the consumers and on that basis, post the relevant advertisements to them.

7.2.1.2 Lack of appropriate inbuilt security mechanisms

Due to the limitation of storage and processing powers, some of the IoT devices which has the provision to implement security features are unable to do so while other have to hardcode the security mechanism within the embedded device. Hard coding the security features is a complex and permanent task. This means that, once coded it cannot be changed. Since the types and nature of threats and vulnerabilities are dynamically changing, hard coding is not considered a safe option for long terms.

7.2.1.3 Unavailability of patching mechanism

In traditional system, once the vulnerabilities and bugs are discovered, they are patched and the system is updated. However this process is fairly intricate and difficult in IoT devices. There are two prime reasons for this:

- Unavailability of appropriate path mechanisms for IoT devices.
- Inability to restart and update the IOT device at multiple times. Because every path requires the system to be rebooted in order to make it effective.

7.2.1.4 Non availability of lightweight security mechanism

The conventional cryptographic techniques are not suitable for the IoT devices. There is an urgent need for lightweight security mechanisms customized for the IoT ecosystem. IoT system requires low powered, extremely fast and secured mechanisms for protecting its data. There are hardly any lightweight security mechanisms available as on date.

7.3 Wireless Sensor Networks

Big data constitutes huge datasets which cannot be effectively handled by classical data management systems and tools. Big data can be seen as a relatively vague term and its definition vary from organization to organization. For example for one organization 100 GB may be termed as big data while for another organization it may be beyond 500 TB of size. The emergence of IoT and its adoption in almost all fields of computing have exponentially increased the generation of data. Although these data are individually very small in size but collectively becomes enormous. The sensors, actuators and transponders constitute the core
elements of the IoT ecosystem ([9], [15], [38], [111], [124], [145], [148], [173]). The interconnections of these devices are governed by wireless sensor networks (WSN). Figure 7.1 presents the architecture of wireless sensor network.

![Layered Architecture of WSN](image)

A WSN may be defined as a collection of huge number of miniature sensors. These sensors are self directed and consume very low power. These sensors can be embedded into network devices or any other equipment or object ([61], [84], [139], [170]). They are responsible for sensing, collecting, processing and transferring the data about its surroundings (like temperature, pressure, humidity, blood pressure, heart beat, pulse rate etc) to the base station or operators. A WSN is highly useful in situation where we cannot create a typical wired network like terrains, sea, underground etc. A WSN can be considered as the backbone of the IoT ecosystem. Figure 7.2 shows an example of a typical WSN. Some of the advantages of WSN are given below:

- Ability to create and manage moveable network.
- Low cost setup.
- No need for complex wiring.
- Easily extensible and scalable.
• Support variety of network topologies.

7.4 RBSEE Architecture [8]

Our proposed RBSEE architecture adopts Atrain Distributed System (ADS) as the base of the architecture [8]. It also uses “Software Defined Networking (SDN)” and “Twofish cryptographic technique”. SDN is incorporated here to provide dynamic network management capabilities while Twofish is used to protect the data stored in the system as well data in transit. For providing energy efficiency, RBSEE introduces the concept of Request-Type Identifying (RTI) unit. Using the RTI unit, the requests are characterized on the basis of certain criteria and parameters, which are explained here in the later section. These components are incorporated in ADS to construct a unified framework for efficiently handling big data generated from IoT devices. The primary aim of RBSEE is to provide a responsive system with an ability to handle requests and queries in real time. The Pilot Computer (PC) of ADS acts as the control unit while the DCs act as IoT devices in the proposed RBSEE architecture. The following categories of requests are handled by RBSEE architecture.

7.4.1 Request Categories in RBSEE

Whenever the user requests for some service to RBSEE, the RTI unit of RBSEE architecture identifies the type of request and categorizes it in one of the following categories:

7.4.1.1 Requests demanding Instant Response
The requests belonging to this category must be responded instantaneously by the IoT system. Few examples of this category include: open the gate, tell the time, make a call, play a song, download a file, switch on the computer etc.

7.4.1.2 Requests demanding Scheduled Response

The requests belonging to this category comprise of those requests that require a scheduled response from the IoT system. i.e. response required after a prefixed interval of time. Few example of this category are: Switch on the geyser at 6.00 a.m every morning, switch off the lights at 11 p.m. every night, switch on the water sprinkler of the garden every evening at 5.00 p.m. etc.

7.4.1.3 Requests demanding Delayed Response

The requests belonging to this category comprise of those requests that do not require instantaneous reply but can be replied once in a day, week or a month at a predefined time.

Few examples of this category are: total calories intake/burnt in a day, total fuel consumed in a day, weekly power consumption, grocery inventory reminders etc.

7.4.1.4 Requests demanding Continuous response

This category comprises of requests that need continuous responses to be sent to the user. Few example of this category are: real time traffic updates, continuous monitoring and measurement of ECG, heart beats, glucose and pulse rate of the patients etc.

7.4.1.5 Event Driven Response

This category of requests comprises of those requests which require responses on the basis of certain conditions as and when the event of interest occurs. They follow IF-A then-B type conditions.

Few example of this category are: If. Outside temperature is hot, lower the temperature of the Air Conditioner, Switch off the lights of the room when everyone has left etc.

7.4.1.6 No Response Required

Requests belonging to this category do not require the response to be given to the users, rather only require the work to be completed.
Few example of this category are: backup data, system upgrades, system scaling etc.

7.4.2 Sensor Categories in RBSEE Approach

RBSEE architecture involves two broad categories of sensors. These are mentioned below:

7.4.2.1 Active Sensors

They are those sensors which become active and remain active with respect to the type of request the user has requested ([8], [196]).

For example, if the user requests for the driving directions from place a to place b, the location finding sensor becomes immediately active from the time the request is given and it remains active till the user has arrived at the destination.

7.4.2.2 Passive Sensors

These kinds of sensors remain in standby mode most of the time. They consume minimum energy as most of the time they remain inactive ([8], [196]). These sensors become active only when the event of interest occurs or the user explicitly requests for some services involving these sensors.

For example the sensor embedded in the geyser become active only when the user gives instruction to start the geyser, while it remains in a standby (sleep mode) mode for the rest of the time.

Each component in the IoT ecosystem is embedded with multiple (active or passive) sensors which are responsible for sensing, monitoring and capturing the physical phenomenon from the surroundings and vicinity of their deployments. These sensors act as individual nodes which are self sustainable and can connect and communicate wirelessly to the base station with the help of a gateway node and relay nodes for transferring the sensed data. The RBSEE architecture uses a combination of active and passive sensors for servicing different types of requests on the basis of the type of request.

7.4.3 Components of RBSEE Architecture

RBSEE architecture is aimed at dynamically servicing the requests of the user on the basis of its type rather than adopting a generalized strategy for servicing the requests. Security, reliability, consistency and energy efficient routing of the data in transit constitute the core characteristics of RBSEE. It provides a generalized framework which can be adopted in any domain of IoT like healthcare, transport, home automations, education etc. Security remains
the primary concern when we talk about adopting IoT technology in any confidentiality specific domain like healthcare or defence. With RBSEE we aim to address this concern along with faster and energy efficient servicing of requests. Figure 7.3 shows the architectural framework of the RBSEE system. It can be observed from figure 7.3, that there are eight components in RBSEE architecture viz User, Request-type Identifying Unit (RTIU), Data Encryption and Standardization unit, System controller, pilot computer, IoT devices, relay nodes and software defined networking unit. The functional description of each of these components is given below.
7.4.3.1 User
They are the individuals or systems that interact with the RBSEE. In order to get any work done, the users request the RBSEE system thorough the user interface to perform the desired operation.

7.4.3.2 Request-Type Identifying Unit (RTIU)
This unit is responsible for differentiating among the different types of requests. The various types of request categories are given below:
(i) Requests demanding Instant Response
(ii) Requests demanding Scheduled Response
(iii) Requests demanding Delayed Response
(iv) Requests demanding Continuous response
(v) Event Driven Response
(vi) Requests where No Response Required
The details about these requests types are already presented in above section.

7.4.3.3 Data Encryption and Standardization (DES) Unit
The DES unit is responsible for encrypting the requests from the users as well as the data captured by the sensors of the IoT devices. This is done to ensure that the data and requests always remain secure whether in transit or at rest in storage devices on the cloud. The data standardization component is responsible for converting the data collected from the IoT devices in a common standard format so that we can perform appropriate and consistent analysis of the data. The process of data standardization is imperative because of the heterogeneity and intricate structure of the associated IoT devices that constitutes the system. In any large IoT ecosystem, the IoT devices that constitute the system are generally manufactured by different vendors and largely differ in make, model, and storage formats. Therefore it is necessary to convert the data captured by these varied devices into a predefined standard format.

7.4.3.4 System Controller
It is the brain of the RBSEE system. It controls every other component by providing instructions to perform as per the need and requirements of the requests from the users.

7.4.3.5 Pilot Computer (PC)
It is the central node which consists of the metadata of all the connected IoT devices of the system. More specifically it stores information like “device ids” of the IoT individual devices, individual data formats, size of data formats, distance of each device from the PC and other important information like respective functionalities of individual IoT devices.

7.4.3.6 IoT Devices
They are miniature devices which consist of the sensors embedded in them. These devices are installed at the desired locations by the users, sometimes they are also used as wearable devices which the user can wear on their body like (smart band, smart watch etc). The sensors embedded in these devices captures the data of the surroundings and forward the data to the base station either continuously or periodically depending upon the type of sensor and requests.

7.4.3.7 Relay Nodes
In order to perform energy efficient routing of data and information from one place to another in RBSEE system, the concept of relay nodes has been incorporated ([61], [115], [173]). These are tiny sensor nodes which act as a forwarding node and are responsible for forwarding the data captured by the neighbouring sensor nodes to the base station in case the sensor node are at a distant place from the base station.

7.4.3.8 Software Defined Networking (SDN) Unit
RBSEE system incorporates SDN in order to dynamically control the network with respect to the traffic and congestion in an IoT ecosystem.

7.4.4 Working of RBSEE Approach
Every IoT device in RBSEE system needs to register itself with the system. This registration is done by sending the device related information like (device_id, processing power, make, model, data format) to the PC. The PC stores this information as a metadata for each device. The registration is essential in order to identify, connect and communicate with the individual IoT device in future references. When the user requests for some service to the RBSEE system, it is first forwarded to the RTIU where it is analyzed to identity the category to which the current request belongs, on the basis of its own type. After this analysis, the current request is categorized in any one of the above explained categories of requests. Furthermore the RTIU attaches a request id (req_id) and category identifier (cat_id) with the incoming request and passes it to the DES unit. The DES unit encrypts the request and converts it into a
predefined format. It then passes the request to the IoT system controller. The request is
decrypted by the IoT system controller. After decrypting the request, it is analyzed and its
category is identified. Then the system controller instructs the PC to identify the appropriate
IoT device(s) to service the request. After identification of appropriate IoT devices, the PC
instructs them to handle the request. The IoT device(s) on receiving the instructions from the
PC performs the desired operation either individually or by integrating the information from
other IoT devices (if required). If a request requires integration of information from multiple
IoT devices, the RBSEE ensures that every individual device is unaware of this integration.
The data integration unit inside the pilot computer uses the pseudonymization technique to
replace the identifying fields of individual IoT devices by artificial identifiers and thus
promote an anonymous information linkage among various participating IoT devices to serve
the user’s request. The transfer of requests, data and information from one unit to another in
RBSEE system is governed by SDN. It is the responsibility of SDN to critically monitor and
analyse the traffic and congestion in the network and choose the optimal route for transferring
these data and requests from one unit to another. The IoT devices that constitute the RBSEE
system are wirelessly connected together with each other and with the PC. Every IoT device
has a direct connection with the PC. The different IoT devices and the PC are connected
together using multi-horse cart topology [30]. When the requested operation is successfully
completed, the sensor embedded in the IoT device(s) notifies the PC. The PC on the other
hand acknowledges the IoT device(s) and prompts the user about the successful completion
of the requested operation (this is done only in case it is an explicit service requested by the
user, otherwise, no notification is sent to the user, only the requested service gets completed).
Figure 7.4 depicts the flowchart presenting the overview of RBSEE approach [8]. The
following algorithm shows the working of RBSEE Architecture:
Algorithm: Working of RBSEE Architecture

1. **FOR** every explicit request from the user, **DO**
2. Identify the request-type and categorise it.
3. Forward it to the DES unit.
4. Convert the request into a predefined standard format.
5. Forward the converted request to IoT System controller.
6. Decrypt the request and analyse it.
7. Instruct the PC to identify the IoT device(s) for performing the requested service.
8. Instruct the identified IoT device(s), to perform the requested Service.
9. **END FOR**
For the sake of understanding let us take one simple example. Suppose that Mr. X has several IoT devices installed at his home automation system. Now Mr. X (user) who was out for work is coming back to home. The proximity sensor and the route map attached to the IoT device in the car senses that the user will reach home in next 15 minutes depending upon the current speed and traffic on the road. It (sensor) immediately communicates with the PC and
forwards this information to it. The PC analyses the information and forwards it to the DES for encrypting it and then forwards it to the cloud storage. After this, it informs the system controller about the information received from the sensors. The system controller fetches the information stored on the cloud storage and after analyzing it, instructs the sensor embedded in the Air-Conditioner to switch it on. Furthermore since the user has a habit of drinking coffee after reaching home. The PC instructs the coffee maker to get the coffee ready. Similarly the PC instructs the geyser to set the optimal temperature for the user to take a bath. As soon as the user reaches home, the sensors embedded in the Gate, open it and the user gets in. Once the user is in front of the main door, the camera embedded on the door performs the facial recognition of the users and opens the gate.

7.4.5 Energy Efficiency in RBSEE Architecture

This section focuses on the energy efficiency of the RBSEE system. As we are aware that every wireless sensor network (WSN) is constituted of several miniature nodes (sensors) which are capable of detecting any event of interest or any change in its surrounding environment and send this data to the base station (or controller node) ([61], [84], [139], [170]). These sensors are generally self-sustained and work without any external interventions. In order to perform their respective functionalities, these sensors consist of miniature battery or alternate energy (e.g. solar energy) source to provide appropriate energy to keep them alive and functioning. The lifetime of the sensors is directly dependent of the types of processing and operations performed as well as the capacity of the battery (energy source). Therefore it is imperative to device novel technologies, techniques and devices which consumes minimal amount of energy without compromising on the processing capabilities and thus, increasing the lifetime of the sensors. The three major factors that affect the amount of energy consumed by the sensors are mentioned below ([8], [9], [15], [38], [111], [124], [145], [148], [173]).

- The distance between the source and destination.

As we are already aware, that a sensor senses the surroundings and forwards this information to the processing node (base station) in the network. The longer the distance between the sensor and the base station, more power is consumed for transferring the data from source to destination.

- Type of activity performed by the sensor
The dissipation of energy (power) is directly proportion to the nature and amount of activity performed by the sensor.

- **Traffic and congestion in the network**

If the network is densely populated, there are chances of increased network traffic and congestion in routes. This makes the sensor wait until a congestion free route is discovered for the data transfer. This unnecessary waiting of the sensors to get the free route also consumes significant amount of energy. This is because of the reason that in most of the proprietary network devices, the data transfer routes are hardcoded within the device firmware and thus are prefixed. This means that the similar types of data travel through the same path irrespective of the network traffic or congestion.

In order to overcome these limiting factors, the RBSEE approach adopts the concept of relay nodes and multi-hop ([9], [15], [38], [84], [91], [94], [111], [124], [145], [148], [173]) communication strategy using SDN to transfer the sensor data from one node to another and to dynamically manage the network configuration in real-time. Figure 7.5 given below shows a direct connection between source and destination.

![Fig 7.5 Direct Communication between source and destination](image)

Relay nodes belong to special category of wireless nodes which are deployed in the WSN to relay the data generated by other sensor nodes to the next node in the network without themselves sensing the environment ([61], [115], [173]). The concept is analogous to the relay race in which we have multiple racers. Each racer covers some distance and passes the baton to the next racer and so on till the last racer reaches the finish point. The advantage of deploying relay nodes in the network is that it can act as a hop between two or more sensor nodes and can shorten the transit distance between two or more sensor nodes. With the help of relay nodes, the energy (power) required by the sensors to transfer the data can be reduced to a large extent. Thus, the relay nodes help in relaxing the over-burdened sensor nodes within the network and consequently play a vital role in enhancing the lifetime of the sensor nodes. Figure 7.6 shows the deployment of relay nodes in a typical WSN.
The multi-hop communication ensures that less power is consumed in communications as the data is transferred from source to destination through closely placed multiple hops instead of using direct link between source and destination which consumes more power ([38], [84], [91], [94]). The appropriate placement of relay nodes in the wireless network is a different class of problem which is not considered in our research work here for this thesis. Figure 7.7 shows a multi-hop communication between source and destination.

Apart from deploying relay nodes for saving the energy, RBSEE approach also ensures that the sensors which are not required for performing the work remains in standby mode and consume the minimal energy required for their survival. Only those sensors which are required for serving critical (immediate responses required) requests are made active and the rest of the sensors in the proposed system remain inactive until the event of interest occurs.

The transfer of data in the RBSEE network is governed by SDN. SDN dynamically updates the routing information and keeps the list of the optimal routes for the data transfer within the network. It is also responsible for managing the network congestion and traffic by dynamically updating the network configuration as and when required in order to maintain a fast, reliable and congestion free network.
7.5 Conclusion

Internet of things (IoT) has given us the concept wherein we can have several types of wireless connections and communication among the constituent objects of the IoT ecosystem. We can have connections and interactions like Thing to Thing (T-to-T), Human to thing (H-to-T), Human to Things (H-to-Ts), Thing to Things (T-to-Ts) and Things to Things (Ts-to-Ts). IoT has brought a paradigm shift and totally transformed the way we visualize and observe a network. The current era of IoT-enabled computing has seen an exponential growth with worldwide adoption and acceptance in the last few years. We can observe that in recent times, more and more objects are getting connected to each other via one or the other means. We have a flood of IoT enabled devices for the modern consumer conveniences. As per Gartner [85], there will be fifty million connected devices by the year 2020. Furthermore, the isolated clouds are also fusing to serve as Fog. In a typical fog computing environment, the computations, storage, interconnection among various participating entities of a network occur at the most suitable location between the cloud data centres and the end-devices ([32-33], [178]).

The world is on the verge of extreme technological advancements and if it is not done in a secured and privacy preserved manner, the result will surely be adverse. Therefore security and privacy must be the primary concern for the development of effective management, deployment and collaboration tools and techniques for creating an efficient IoT ecosystem. The RBSEE system [8] takes care of the security and privacy concerns of the data captured and transmitted by the sensors. Furthermore with the adoption of SDN, the network configuration and management is dynamically controlled by the RBSEE administrator which enables the system to choose appropriate route for transfer of data from one node to another within the network. The RBSEE approach provides energy efficiency by incorporating relay nodes and multi-hop communications along with selectively notifying the sensors to wakeup only when the event of interest occurs.