E-TOLL PAYMENT USING AZURE CLOUD

M Vikram[1], M R Rajeswari[2], V Veeralakshmi[3]

[1]*Student & Dept. of Information Technology, Prince Shri Venkateshwara Padmavathy Engineering College, Chennai, India*
[2]*Assistant Professor & Dept. of Information Technology, Prince Shri Venkateshwara Padmavathy Engineering College, Chennai, India*
[3]*Associate Professor, Dept. of Information Technology, Prince Shri Venkateshwara Padmavathy Engineering College Chennai, India*

---***---

Abstract- *Internet of Things is the inter-working of physical devices, vehicles which is referred as smart devices, building and other items embedded with electronics, cloud, software, sensors, actuators and network connectivity that enables the objects to collect and exchange data The IoT allows objects to be sensed or controlled remotely across existing network infrastructure, creating opportunities for most direct integration of the physical world into computer-based systems, and resulting in improved efficiency, accuracy and economic benefit in addition to reduced human intervention. Three of the main concerns that accompany the Internet of Things are the breach of privacy, over-reliance on technology and the loss of jobs. There are security measures that are taken to protect information, but there is always possibility of hackers breaking into the system and stealing the data. Toll Gate Payment application have been of great assistance in lessening the over congestion that has become a part of the metropolitan cities these days. It is one of the uncomplicated ways to manage the great run of traffic. The travelers passing through this mode of transport, carried by their transport that allows them to be aware of the account of money that has been paid and the money left in the tag. It relieves the traveler of the burden of waiting in the queue to make the toll payment, which decreases the fuel-consumption and taking cash with them can be avoided. Here, it avoids this type of problem. User get gate pass by recognize the vehicle registration number using RFID so user don't need to wait in tollgate.*

Key Words: RFID tag, Cloud, Sensor, Security, Raspberry Pi

1.INTRODUCTION

The Internet of Things (IoT) is an important topic in technology industry, policy, and engineering circles and has become headline news in both the specialty press and the popular media. This technology is embodied in a wide spectrum of networked products, systems, and sensors, which take advantage of advancements in computing power, electronics miniaturization, and network interconnections to offer new capabilities not previously possible. An abundance of conferences, reports, and news articles discuss and debate the prospective impact of the "IoT revolution"—from new market opportunities and business models to concerns about security, privacy, and technical interoperability.

The large-scale implementation of IoT devices promises to transform many aspects of the way we live. For consumers, new IoT products like Internet-enabled appliances, home automation components, and energy management devices are moving us toward a vision of the "smart home", offering more security and energy efficiency. Other personal IoT devices like wearable fitness and health monitoring devices and network enabled medical devices are transforming the way healthcare services are delivered. This technology promises to be beneficial for people with disabilities and the elderly, enabling improved levels of independence and quality of life at a reasonable cost. [1] IoT systems like networked vehicles, intelligent traffic systems, and sensors embedded in roads and bridges move us closer to the idea of "smart cities", which help minimize congestion and energy consumption. IoT technology offers the possibility to transform agriculture, industry and energy production and distribution by increasing the availability of information along the value chain of production using networked sensors. However, IoT raises many issues and challenges that need to be considered and addressed for potential benefits to be realized.

The Internet Society cares about the IoT as it represents a growing aspect of how people and institutions are likely to interact with the Internet in their personal, social, and economic lives. If even modest projections are correct, am explosion of IoT applications could present a fundamental shift in how users engage with and are impacted by the Internet, raising new issues and different dimensions of existing challenges across consumers concerns, technology, policy and law. IoT also will likely have varying consequences in different economies and regions, bringing a diverse set of opportunities and challenges across the globe.

The concept of combining computers and networks to monitors and control devices has been around for decades. The advancement in wireless technology allowed "machine to machine" (M2M) enterprise and industrial solutions for equipment monitoring and operation to become widespread.

Manu of these early M2M solutions, however, were based on closed purpose-built networks and proprietary or industry-specific standards [2], rather than on Internet Protocol(IP) based networks and Internet standards.

Using IP to connect devices other than computers to the Internet is not a new idea. The first Internet "device" an IP enabled toaster that could be turned on and off over the Internet was featured at an Internet conference in 1990. [3] Over the next several years, other "things" were IP enabled, including a soda machine [4] at Carnegie Mellon University in the US and a coffee pot [5] in the Trojan Room at the University if Cambridge in the UK. From these whimsical beginnings, a robust field of research and development into "smart object networking" [6] helped create the foundation for today's Internet of Things. The users of IoT devices are likely to be more concerned with the services delivered and implication of using those services than issues of when or where data passes through an IP based network.

2. INTERNET OF THINGS COMMUNICATION MODELS

From an operational perspective, it is useful to think about how IoT devices connect and communicate in terms of their technical communication model. In March 2015, The Internet Architecture Board(IAB) released a guiding architectural document for networking of smart objects (RFC 7452) [7], which outlines a framework of four common communication models used by IoT devices. The discussion below presents this framework and explains key characteristics of each models in the framework.

2.1 Device to Deivce Communication Model

Wherever Times is specified, Times Roman or Times New Roman may be used. If neither is available on your word processor, please use the font closest in appearance to Times. Avoid using bit-mapped fonts if possible. True-Type 1 or Open Type fonts are preferred. Please embed symbol fonts, as well, for math, etc. The device to device communication model represents two or more devices that directly connect and communicate between one another, rather than through an intermediary application server. These devices communicate over many types of networks, including IP networks or the Internet. Often, however these devices use protocols like Bluetooth, [8][9] Z-Wave, [10] ZigBee [11] to establish direct device to device communication, as shown in Figure 1.

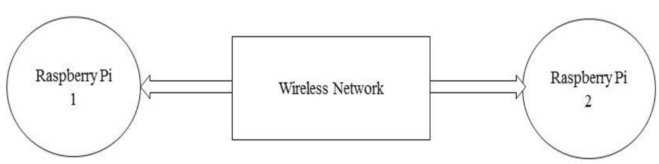

Fig. 1 Device to Device communication

These devices to device networks allow devices that adhere to a communication protocol to communicate and exchange messages to achieve their function. This communication model is commonly used in application lie home automation systems, which typically use small data packets of information to communicate between devices with relatively low data requirements. Residential IoT devices like light bulbs, light switches thermostats, and door locks normally send small amounts of information to each other in a home automation scenario.

2.2 Device to Cloud communication model

In a device to cloud communication model, the IoT device connects directly to an Internet Cloud service like an application service provider to exchange data and control message traffic. This approach frequently takes advantages of existing communication mechanism like traditional wired Ethernet or Wi-Fi connections to establish a connection between the device and the IP network, which ultimately connects to the cloud service. This is shown in Figure 2.

This communication model is employed by some popular consumer IoT devices like the Nest Labs Learning Thermostat. [12] and the Samsung SmartTV[13]. In the case of the Nest Learning Thermostat, the device transmits data to a cloud database where the data can be used to analyze home energy consumption. Further, this cloud connection enables the user to obtain remote access to their thermostat via a smartphone or Web interface, and it also supports software updates to the thermostat. Similarly, with the Samsung SmartTV technology, the television uses an Internet connection to transmit user viewing information to Samsung for analysis and enables the interactive voice recognition features of the TV. In these cases, the device to cloud model adds value to the end user by extending the capabilities of the device beyond its native features.

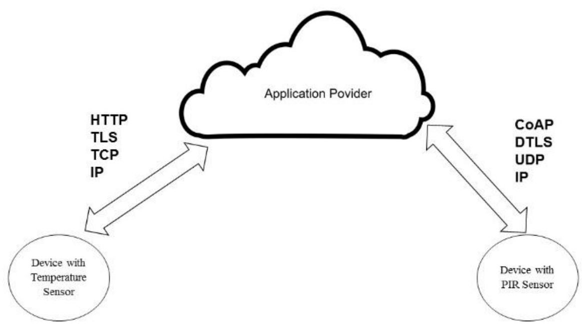

Fig.2 Device to Cloud Communication model diagram.

2.3 Device to Gateway communication model

In the device to gateway model, or more typically, the device to application layer gateway(ALG) model, the IoT device connects through an ALG service as a conduit to reach a cloud service. In simpler terms, this means that there is application software operating on local gateway device, which acts as an intermediary between the device and cloud service and provides security and other functionality such as data or protocol translation. The model is shown in Figure 3.

Several forms of this model are found in consumer devices. In many cases, the local gateway device is a smartphone running an app to communicate with a device and relay data to a cloud service. This is often the model employed with popular consumer items like personal fitness tracker. These devices do not have the native ability to connect directly to a cloud service, so they frequently rely on smartphone app software to serve as an intermediary gateway to connect the fitness device to the cloud. The other form of this device to gateway model is the emergence of "hub" device in home automation application. These are devices that serve as a local gateway between individual IoT devices and a cloud service, but they can also bridge the interoperability gap between devices themselves.

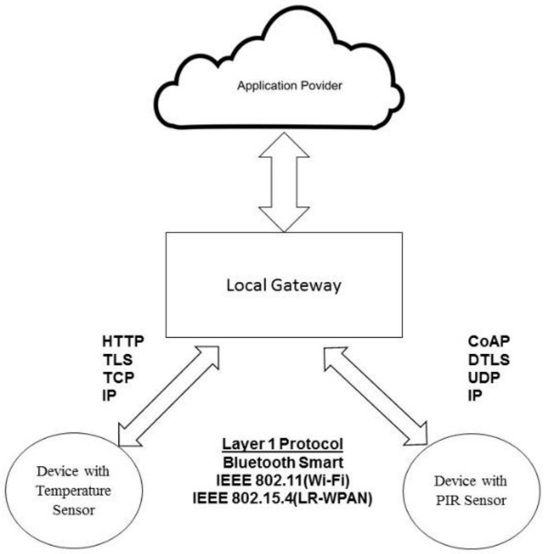

Fig. 3 Device to Gateway communication model diagram

2.4 Back-End Data Sharing Model

The back-end data sharing model refers to a communication architecture that enables users to export and analyze smart object data from a cloud service in combination with data from other sources. This architecture supports "the user's desire for granting access to the uploaded sensor data to third parties". [15] This approach is an extension of the single device to cloud communication model, which can lead to data silos where "IoT devices upload data only to a single application service provider". A back-end sharing architecture allows the data collected from single IoT device data streams to be aggregated and analyzed. The back-end data sharing model suggests a federated cloud services approach or cloud applications programmer interfaces(APIs) are needed to achieve interoperability of smart device data hosted in the cloud. [16] A graphical representation of this design is shown in Figure 4.

Often in the single device to cloud model, the data each IoT sensor or system produces sits in a stand-alone data silo. An effective back-end data sharing architecture would allow the company to easily access and analyze the data in the cloud produced by the whole spectrum of devices in the building. This kind of architecture facilitates data portability needs. Effective back-end data-sharing

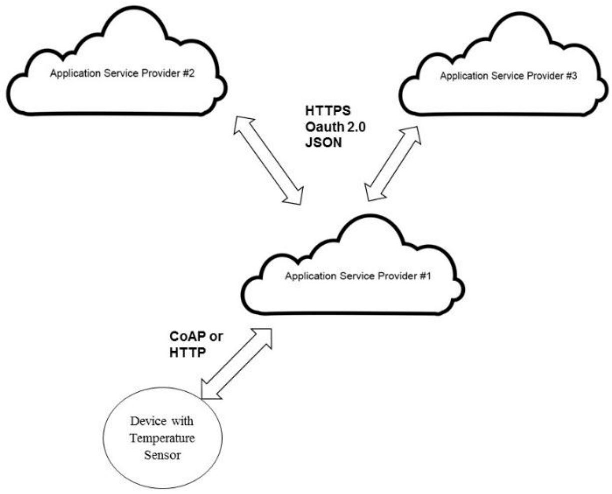

Fig. 4 Back-end data sharing model

which deals with Cloud, Internet and Mobile Application to architectures allow users to move their data when they switch between IoT services, breaking down traditional data silo barriers. In this communication model is only as effective as the underlying IoT system design. Back-end data sharing architectures cannot fully overcome closed system designs.

3. RELATED WORKS

Kurti Sanghvi, Prof.Amol Jolekar [17] proposed that the Automating the payment of Toll tax at toll plaza that the money will be deducted the money from the account that is associated with the vehicle registration number automatically. The toll gate will open only if the payment is successful. The process which they proposed is a time consuming one because the user must wait in the queue for their turn. Sometimes the authentication for the account may fail. The main drawback of this methodology is that if there is no money in that specified account then the payment cannot be done and vehicle cannot pass through.

V Sridhar and M Nagendra[18] focused on the smart card based toll automated system use of smart card which is nothing but like a memory card in which we are going to store the details of person and certain amount. The main objective of this smart card is to pay the toll gate tax using smart card. Smart card must be recharged with some amount and whenever a person wants to pay the toll gate tax, he needs to insert his smart card and deduct amount using keypad. By using this kind of smart card there is no need to carry the amount in form of cash and so we can have security as well. Keil-µ vision software used for simulation. Integrating features of all the hardware components used have developed it. Presence of every module has been reasoned out and placed carefully thus contributing to the best working of the unit.

Elmer R Magsino and Ivan W. H. Ho [19] proposed that intelligent tollgate queue selector for improving server utilization and vehicle waiting time. The incorporation of a fuzzy logic controller in a highway tollgate system can effectively decrease a vehicle's average waiting time and average queue length, and improve the server's utilization. This is true for both homogeneous and nonhomogeneous vehicular arrivals in the highway lanes especially with tollgates of fast service time. As an added feature to the FLCQ selector, an early warning signal was introduced to allow the server to reduce its service time when the traffic is building up due to relatively long service times. This work can be further extended by including a practical modeling of the service time reduction and the scalability of servers, i.e. deactivation of existing servers or addition of needed servers based on existing infrastructure. The nonhomogeneous service time of manual servers can also be taken into consideration. Finally, allowing communications between vehicles and highway infrastructure of patterns/features detected by camera or sensors will further develop a more intelligent and highly adaptive highway tollgate system in the future.

4. PROPOSED SYSTEM

IoT is an effective technology to solve the real-time problems and automating the process. IoT is a technology which deals with Cloud, Internet and Mobile Application to produce a solution. In the is paper each process is automated and the decisions are taken in the Cloud with the data obtained from the Raspberry Pi and the reference data obtained from the mobile after making the payment. In this paper, the payment for the toll gate is done through the mobile application named E-TOLL. Payment is done using the digital wallets, credit card, debit card and net banking. Once the payment is successful the immediately the vehicle registration number and transaction id will be forwarded to the Azure Cloud and it will be stored as a reference data. The RFID (Radio Frequency Identifier) tag will be attached with the vehicle and the vehicle registration number will be embedded into the RFID tag. When the vehicle reaches, the toll gate the RFID reader will automatically reads the vehicle registration number which is embedded in the RFID tag. Then the vehicle registration number will be forwarded to the cloud and checks whether the payment is done or not. If the payment is done the response will be sent to the device and the action will be taken depending upon the response from the Cloud. If the payment is done, then the tollgate will open. Otherwise, the toll gate will not open.

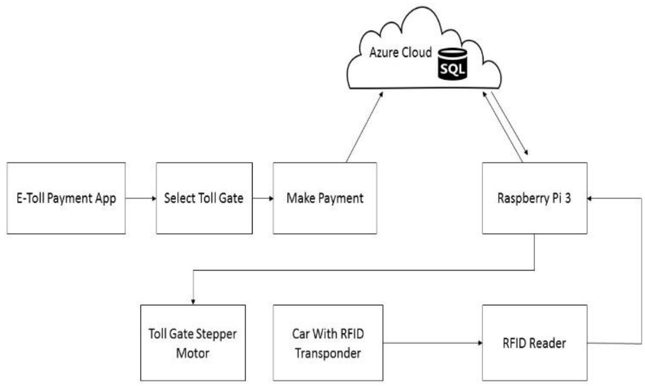

Fig. 5 Architecture of E-Toll Payment using Azure Cloud

A graphical representation of the architecture of E-Toll Payment using Azure is shown in the Figure 5. A system architecture is the conceptual design that defines the structure and behavior of a system. An architecture description is a formal description of a system organized in a way that supports reasoning about the structural properties of the system. An Automatic Toll Collection system can determine if a car is registered in a toll payment program, by comparing the unique code given to each car that is stuck to the windshield of the car, with the database stored in the processor to authenticate the incoming car. After the cars are authenticated and they match the unique ID stored in the database, the respective toll amount gets deduced from their bank accounts following which the gate automatically opens and the cars can pass else they are not allowed. Electronic toll collection should be made a globally accepted method of toll collection, a trend greatly aided by the advancement in the field of interoperable Electronic Toll Collection technologies. When the user has entered the source and destination details, the application will automatically show the tolls which are present in the travelling route and the toll amount also. The user can select the tolls to which they want to make the payment. The user can select as many toll gates as they want. According to tolls they have selected the user has to make the payment. The payment can be done using the different digital wallets or credit card or debit card or net banking. Once the payment is successful, then immediately the transaction id and vehicle registration number will be transmitted to the Azure cloud. The transaction id and vehicle registration number will be stored in the blob in the storage account. The blob is one of the storage type in the storage account. The blob is considered to the reference data and the file in the blob will be periodically updated.

The RFID (Radio Frequency Identifier) receiver will be integrated with the raspberry pi device. The RFID receiver will receive the data transmitted by the RFID transponder. The RFID transponder will be attached to the vehicle and the transponder will be written with the vehicle registration number. The transponder uses the WORM (Write Once Read Many) technique to write the vehicle registration number. When the RFID receiver receive the data, its being forwarded to the raspberry pi device and the raspberry pi device forwards the data to the azure cloud through the MQTT (Message Queue Telemetry Transport). The raspberry device will connect with the IoT hub in the Azure cloud. A single IoT hub can connect to 500 devices. The IoT hub forwards the data to the Stream Analysis. The Stream Analysis has two inputs. One is the reference data and the other is the data stream. The reference data contains the data transmitted from the mobile application and the data stream consist of the data from raspberry pi. The query will be written to fetch the necessary information from the two input and give the output. The output will be the IoT hub where the data is again sent to the device from the device it received the data. And if the received data is true then the toll gate will open then the user can leave the toll. In case the received data is false then the toll gate will not open unless the user has made the payment. Every day the scheduler will automatically create the necessary file and backup the data at the end of the day.

4.1 Mobile Application

The payment for the toll gates are done using the mobile application which has been created. The mobile application has been designed in the android platform. The mobile application uses the Azure cloud database server for storing and retrieving the data from the Cloud. The salt hashing technique is used to hash the password that is being entered during the registration in the mobile application. Whenever the user tries to login, the password entered is converted into hash value and checked with hash value stored in the database. If the validation is successful then the user is logged in, otherwise the user cannot login into their account. The Figure 6 displays the screen where the user enters the travel details like source, destination and the vehicle type. Based on the vehicle type the rate for the toll amount is being calculated.

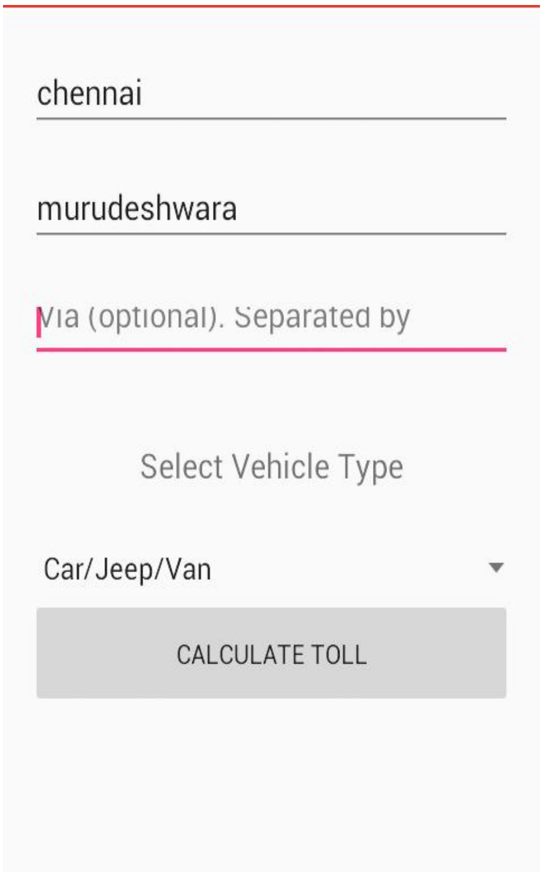

Fig 6. Mobile Application Travel Details Screen

The Figure 7 shows the available toll gate between the source and destination. The rate for all the available toll gate and for the specific vehicle is also listed in the Toll Gate List Screen. All the toll gate details will be stored in the server. The request is sent to the server via JSON (JavaScript Object Notation) and then only the details are listed in the mobile application. The entire amount for the toll gate will be calculated in the background.

Fig. 7 Mobile Application Toll Gate List Screen

5. CONCLUSIONS

The concept of the E-Toll Payment using Azure Cloud is presented. The recent technological application that makes the human life more beneficial in the present world have been discussed. The new standard of Internet of Things is already is in competition with each other is also one of the talk of the topic in near future when other more standards are coming soon. The E-Toll Payment application using Azure will reduce the amount of time waiting in the queue to pay for the toll gate. They can make the payment through the digital wallets or Internet banking or different options. The payment can be done for the multiple toll gate and the records can be easily maintained.

REFERENCES

[1] For more information on IoT as it relates with those with disabilities see for example: Valerio, Pablo. “Google: IoT Can Help The Disabled.” InformationWeek, March 10, 2015. http://www.informationweek.com/mobile/mobile-devices/google-iot-can-help-thedisabled/a/d-id/1319404; and, Domingo, Mari Carmen. “An Overview of the Internet of Things for People with Disabilities.” Journal of Network and Computer Applications 35, no. 2 (March 2012): 584–96. doi:10.1016/j.jnca.2011.10.015. M. Young, The Technical Writer’s Handbook. Mill Valley, CA: University Science, 1989.

[2] Polsonetti, Chantal. “Know the Difference Between IoT and M2M.” Automation World, July 15, 2014. http://www.automationworld.com/cloud-computing/know-difference-between-iot-and-m2m I. S. Jacobs and C. P. Bean, “Fine particles, thin films and exchange anisotropy,” in Magnetism, vol. III, G. T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271–350.

[3] "The Internet Toaster." Living Internet, 7 Jan. 2000. Web. 06 Sept. 2015. http://www.livinginternet.com/i/ia_myths_toast.htm

[4] "The "Only" Coke Machine on the Internet." Carnegie Mellon University Computer Science Department, n.d. Web. 06 Sept. 2015. https://www.cs.cmu.edu/~coke/history_long.txt.

[5] Stafford-Fraser, Quentin. "The Trojan Room Coffee Pot." N.p., May 1995. Web. 06 Sept. 2015.

[6] RFC 7452, “Architectural Considerations in Smart Object Networking” (March 2015), https://tools.ietf.org/html/rfc7452.

[7] Tschofenig, H., et. al., Architectural Considerations in Smart Object Networking. Tech. no. RFC 7452. Internet Architecture Board, Mar. 2015. Web. https://www.rfc-editor.org/rfc/rfc7452.txt.

[8] http://www.bluetooth.com.

[9] http://www.bluetooth.org.

[10] http://www.z-wave.com.

[11] http://www.zigbee.org.

[12] “Meet the Nest Thermostat | Nest.” Nest Labs. Web. 31 Aug. 2015. https://nest.com/thermostat/meet-nest-thermostat/.

[13] “Samsung Privacy Policy--SmartTV Supplement.” Samsung Corp. Web. 29 Sept. 2015. http://www.samsung.com/sg/info/privacy/smarttv.html

[14] “How It Works.” SmartThings, 2015. http://www.smartthings.com/how-it-works.

[15] Tschofenig, H., et. al., p. 9.

[16] An example of a generic (non-IoT) off-the-shelf, federated cloud-sharing tool is ownCloud, produced by ownCloud.org. https://owncloud.org/blog/faster-easier-file-sync-and-share-with-federated-self-hosted-owncloud-8-0/.

[17] Kurti Sanghvi and Prof. Amol Joglekar, “Automating the Paymenr of Toll Tax at Toll Payment”, International Journal of Cimputer Science and Information Technologies, Vol. 6(3), 2015.

[18] V Sridhar and M Nagendra, “ Smart Card Based Toll Gate Automated System”, International Journal if Advanced Research in Computer Engineering and Technology, Volume 1, Issue 5, July 2012.

[19] Elmer R. Magsino and Ivan W. H. Ho, “An Intelligent Highway Tollgate Queue Selector for Improving Server Utilization and Vehicle Waiting Time’, 2016 IEEE Region 10 Symposium (TENSYMP), Bali, Indonesia.