Journal of
Data Acquisition and Processing
ISSN: 1004-9037
https://sjcjycl.cn/
DOI: 10.5281/zenodo.7766105

BUILDING SECURE CLOUD-BASED IOT SYSTEM FOR HOME AND INDUSTRIAL AUTOMATION

Annu Nagar
Student, M.Tech (Network Security), Department of Computer Science & Engineering and Information Technology, Bhagat Phool Singh Mahila Vishwavidyalaya, Khanpur Kalan, Sonipat, Haryana, India, Email: annunagar351@gmail.com

Dr. Vinod Saroha
Assistant Professor, M.Tech (Network Security), Department of Computer Science & Engineering and Information Technology, Bhagat Phool Singh Mahila Vishwavidyalaya Khanpur Kalan, Sonipat, Haryana, India, Email: vnd.saroha@gmail.com

Mrs. Saya Barak
Assistant Professor, M.Tech (Network Security), Department of Computer Science & Engineering and Information Technology, Bhagat Phool Singh Mahila Vishwavidyalaya Khanpur Kalan, Sonipat, Haryana, India, Email: saya.dhull@gmail.com

Abstract
The IoT's rapid growth in the realm of home and industrial automation suggests it will soon affect every market. When considering practical ways to find an IoT interface, a cloud server

is naturally the first item to consider. A real-time Internet of Things interface requires a lot of work, but there are benefits to doing it in the cloud. Automating factories and homes requires an internet communication protocol that is both secure and efficient enough to prevent data loss. Because of its low cost of entry, good Quality of Service, and fast data transfer, MQTT protocol is the best option for meeting the demands of the Internet of Things. In today's market, there are several IoT cloud providers, each with the goal of giving their clients the most useful and accurate IoT-based services available. In spite of the pervasive nature of these IoT clouds, we have not initiated the process of regularization on any of the datasets, let alone even a small number of studies that are analytically fairly comparable to one another. Even a tiny number of these studies have not even been considered. Because various Internet of Things applications have diverse needs for cloud resources, it is essential to perform research on IoT clouds. Latency, update frequency, simplicity of use, compatibility with IFTTT, data processing, storage capacity, number of servers, and security may be among these needs; nevertheless, this list is not exhaustive. The purpose of this research was to evaluate and contrast five of the most popular cloud computing platforms, namely Adafruit IO, Amazon Web Services (AWS), Blynk, Thingspeak, and Ubidots, in order to determine which of these platforms is best suited for a specific activity or category of applications. The five platforms that were investigated were: Adafruit IO, Amazon Web Services (AWS), Blynk, and Thingspeak.
Keywords: Internet of Things (IoT), Message Queuing Telemetry Transport (MQTT), IFTTT (If This Then That).

Introduction

The Internet of Things, more commonly referred to as "IoT" for its shortened form, is the expansion of the purview of the internet such that it encompasses not only individual computers and mobile phones, but also a wide range of additional devices, processes, and ecosystems. This broadening of the internet's purview is commonly referred to as "the Internet of Things." Information is generated by components of the system that are "hooked up" to it, and this information is then sent back into the system. Because of their ability to carry out activities utilizing embedded technologies to detect, communicate, interact, and collaborate with objects, home and industrial automation require high-quality connections to the internet of things. This is because of their capacity to automate homes and factories (IoT). This is because these connections have the ability to generate a network of physical objects that is an exceptional illustration of the criterion. The reason for this is due to the fact that these connections have the capacity to form a network. Home automation and industrial automation are two key categories that may be used to describe the applications; each has its own set of requirements, technologies, innovations, and designs, as well as its own degree of complexity. Home automation and industrial automation are two key categories that may be used to describe the applications. The applications are able to be arranged in a variety of categories at one's discretion. In the present day and age, the great majority of Internet of Things applications are carried out on a range of cloud platforms in order to get the most of the benefits that are made available by these settings. We could interpret cloud with MQTT as a system that revolves around the "Cloud Computing" base to achieve bulk handling ability & data to hold its activity that measures all computation, sensing, and data merged into a simple independent IoT cloud server. This would allow us to achieve our goal of bulk handling ability & data to hold our activity. This would make it possible for us to accomplish our aim of being able to handle massive amounts of data and activities. This would make it feasible for us to accomplish our goals of achieving the ability to handle large volumes of data as well as keeping our activities under control. When it comes to the process of gathering communications, maintaining records, and drawing statistical conclusions from the data, the cloud plays a very significant role. In order to do this, it connects to a cloud server and transmits data from all of the controls and sensors that are included inside the device. After arriving at that location, the data are gathered and processed in an efficient manner. As a result of this, cloud computing has been the motivation behind a number of extremely important developments in the field of application development for the Internet of Things. There are several clouds on the internet, and each one of them is working very hard to find new customers and active methods in order to solidify their position as the most important participant in the internet of things (IoT). Despite the fact that the majority of these clouds are primarily focused on showcasing their one-of-a-kind characteristics and capabilities, it seems that the majority of these clouds are less concerned with the quality of the services they provide and the industry standards they adhere to. Rather than making it possible for users to control the resources and capabilities of the cloud, several cloud platforms place a greater focus on educating users in the appropriate administration and operation of their own cloud platforms rather than making it possible for users to control those resources and capabilities. The open rules, standards, and regulations that are managed by the administrators of the clouds have made them to be quite unique from one another. This is the case because of the nature of the rules. As a direct result of this, the clouds can be easily differentiated from one another. Before using a service offered by any of the numerous clouds that are already available on the internet for one's very own internet of things application, it is of the utmost importance for one to become familiar with all of the features, requirements, resources, and characteristics of the specific cloud in question. This is especially true in the case of the particular cloud in question. This is due to the fact that it is very crucial to carry out these steps. There are certain clouds that give their services for absolutely no cost at all, whilst other clouds offer their services for either a one-time payment or on a subscription basis and charge additional fees for either option.

An application designed for the Internet of Things will send queries to a specific cloud, asking for its different resources. This highlights the necessity for a study on the Internet of Things cloud in relation to latency, intervals for future updates, user friendliness, IFTTT compatibility, data handling, processing data, storage limitations, servers utilized, and the level of safety. These features are sufficient to investigate, assess, and acquire an understanding of the services, functions, support, and resources that a certain cloud provides. As a result of this, a comparison is offered on the most recent versions of the clouds (January, 2020), and each of these versions is compared to the other versions in order to establish their capabilities, features, and recommendations. This study, which is based on the criteria that were discussed in an earlier presentation, explores the many characteristics that were formed in five of the most prominent clouds (Adafruit IO, AWS, Blynk, and Thingspeak and Ubidots). The user has the freedom to evaluate any cloud service of their choosing, and they may do so by contrasting and comparing different cloud services with the use of a set of evaluative cloud variables that are likewise entirely up to them to choose. The user is not obligated to limit their attention to just these five clouds or the features that have been highlighted in this paragraph. Users will have the ability to embrace cloud computing in accordance with their needs, which will ultimately result in the standardization of standards and an improvement in service quality. The end objective is to give exact knowledge about the current cloud service providers for the Internet of Things, including the pros and downsides of each.

Literature Review

- **Utilizing mobile phones as a component of an automation system for the house that is based on Bluetooth:**

In a home automation system that is based on Bluetooth, the different household appliances are connected to the Arduino BT board via the relevant input/output ports by means of relays. The software that runs on the Arduino BT board is written in a high-level, interactive version of the language C that is used to program microcontrollers. Additionally, the board makes use of Bluetooth in order to connect to other devices. Because of the password protection, the piece of equipment may only be accessed by those individuals who have been expressly granted permission to do so. Establishing a Bluetooth connection between the Arduino BT board and the phone is the first step that has to be taken before any kind of wireless communication can take place. Python is the programming language that is used by this system. In addition, the script is portable and may be deployed in any of the contexts provided by the Symbian OS. One of the circuits is going to be used to receive feedback from the phone, which will indicate the current condition of the device. This is something that is going to be planned out and carried out.

- **Making use of mobile phones as components of a home automation system based on Zigbee:**

In order for the system to be able to monitor and take control of the appliances in the home, the design and implementation of the system both make use of Zigbee. The network coordinators keep an eye on the performance of the device and make notes about it as necessary. A Wi-Fi network is used in order to accomplish this goal. This particular wireless network makes use of a modern standard wireless ADSL router that has four ports. Both the network's SSID and the Wi-Fi security parameter have previously been set up and established. The message, which for reasons of security must first be processed by the virtual house algorithm, is then re-encrypted before being delivered to the real network device that is situated within the home once it has been determined to be secure. Zigbee controller connected with the end device by way of the Zigbee network by sending messages to it. The safeguarding and maintaining of the secrecy of each and every communication that is brought into the virtual home by the algorithm that governs it. Zigbee connectivity might be useful in lowering both the cost of the system and the level of invasiveness required for each installation of the system.

EXPLORATION AND HYPOTHESIZING

The modern Internet of Things device (ESP8266 NodeMCU) that is connected to the 2.4 GHz Wi-Fi network gets its power from a cellular network that supports 4G LTE and has an internet connection speed of 2 Mbps. Processing times for hardware and networks are disregarded since they are measured in microseconds, which are insignificantly small amounts of time. In order to get results that could be reproduced with high levels of accuracy, the processes were drawn out over unusually long time periods. The only "variable affected is the server and related settings," given that all other aspects, such as physical location, network connection, hardware equipment, and CPUs, remain identical. The server and its associated settings are the only things that have been modified. If such factors are altered, the results may be affected as a direct or indirect consequence. When cloud and hardware equipment desire to exchange the data in a continuous manner, there are two delays that come into play. These delays are known as cloud latency and update delay. The amount of time it takes for data to go to the cloud as well as the amount of time it takes for the cloud to carry out the appropriate action is what causes cloud latency. Due to the fact that data has to travel to a designated point (the feed location in the user's account), there may be a delay in the transmission of equipment updates to the cloud channel. The amount of work that must be done on the server in order to correctly identify the user, decrypt the data, and update the relevant feed is directly related to the amount of time that must be spent on the task. The sum of these two delays is what's referred to as the cloud latency, and it's used to determine the first delay. The second latency happens every time the Internet of Things device uploads fresh data to a different stream in the cloud. There is a delay in updates for IoT devices because service providers have to take safeguards to protect their servers from being overloaded by a sudden surge of data from a variety of devices. Because each server's "client handling and administrative capabilities" determine the length of the upload delay intervals, these times are consistent across all servers.

Throughput is something that has to be taken into consideration in order to have a working IoT cloud. If packets are lost while being sent, the whole Internet of Things might be rendered inoperable. As a consequence of this, the throughput of each cloud must be maximized in order for it to be utilized to its maximum potential. The rate at which the Internet of Things cloud develops is also contingent upon other elements, one of which being the amount of time that passes before connected devices get software upgrades. When a little quantity of data is uploaded at a specified interval, cloud computing moves at a snail's pace; on the other hand, when a huge amount of data is posted at a predetermined interval, cloud computing moves swiftly. Even while simplicity of use may not be a technical problem, it is nonetheless beneficial to both the user and the Internet of Things in terms of their common knowledge, application, and use of cloud services. Users may not be interested in utilizing the service if it is difficult to navigate the cloud, regardless of how powerful the underlying technology may be. Although it is unclear what direct benefits IFTTT contributes to the Internet of Things, the fact that it makes it easier to include other services that are already available online makes it a component that must be taken into consideration. If a user is not provided with sufficient time and storage capacity to back up his data, that data might be lost. As a result, it is essential to provide users with appropriate time and space to save their data and retrieve it when required.

Because these devices often have less processing capacity, many users are drawn to IoT because it offers them the opportunity to save money while also allowing them to make advantage of the cloud's computational potential. This allows the users to upload their primary data to the cloud. The protection of the user's data and the elimination of any interference from a third party with the user's cloud or services are necessary for the user to even consider the Internet of Things (IoT). As a result, the reader will have access to all of the knowledge that she or he needs to properly comprehend the significance that these components have for the Internet of Things as a result of reading this article.

Methodology

Performing Calculations on Cloud Latencies

Cloud latency is the amount of time it takes for data to get mirrored on the cloud after being uploaded from hardware equipment or vice versa. This may be described as the amount of time it takes for the data to become reflected on the cloud. Certain clouds have the capability to manage the channels either by using a mechanism that is driven by interrupts or by detecting the emergence of a data fluctuation in the fields. Both of these methods may be used in conjunction with one another. Both of these approaches have the same amount of merit. The fact that the cloud service provider provides time stamps on a secondly basis significantly helps to increase the accuracy of the latency assessment. Each of the cloud's nodes receives a brand new time stamp one second after it has been generated by the cloud itself. This procedure occurs again and over again without stopping. As a result of this, it is possible to calculate the cloud latency by adding the amount of time that it took for the first packet to travel in a single direction to the amount of time that it took for each subsequent packet to reach its destination in relation to the time that the initial packet arrived. This allows one to determine the total amount of time that it took for the first packet to travel in a single direction. Given that the ping value is comparable to twice the travel time in a single direction, it is easy to compute the travel time of the first packet by reducing the ping value in half. This is because the ping value is equivalent to double the travel time in a single direction. When a time stamp is received, a clock is started, and the value of the time stamp is subtracted from the value of the clock that was used to establish the initial time stamp. This occurs anytime a time stamp is received. The computation will need to be done more slowly if you choose this strategy. There are certain clouds that choose not to participate in the acknowledgment driver procedure. Before a device connected to one of these clouds may get the information it needs, the hardware connected to that cloud must first communicate its wish to receive explicit data from the cloud. These clouds are the only ones that can be recognized due to the fact that they did not participate in the acknowledgment driver procedure. When a request is made and when data is received, the current local time is recorded by a timer using a timer that is used to record the time. When any of those events takes happened, the timer begins counting down from the current time. By deducting the times that are shown on both of the clocks that are displayed on the timer, it is possible to calculate the cloud latency.

User Friendly

The user friendliness of cloud computing is a non-quantifiable but critically important aspect that is defined by the cloud's practical usage, ease of access to resources, display of messages, and ability to provide online help to the customer. Cloud computing's user friendliness is defined by the cloud's practical usage, ease of access to resources, display of messages, and ability to provide online help to the customer. The user friendliness of cloud computing may be described by the cloud's practical use, the simplicity with which users can get access to resources, the display of messages, and the capacity to provide online assistance to customers. Performing computations through the cloud The amount to which a cloud computing platform can be put to practical use, allow simple access to resources, clearly display messages, and provide online support to clients is one way in which its user friendliness may be judged. The adaptability of the approach taken to modify the cloud's resources, the feed types, and the dashboard that has been approved by the cloud server are all essential aspects that are investigated in order to determine the value of the cloud. The flexibility with which the user can design widgets, the ease with which the user can administer widgets, and the adaptability of the approach taken to modify the cloud's resources are all essential aspects that are investigated. A score is assigned to the cloud depending on how well it satisfies the conditions listed above.

IFTTT compatibility

IFTTT is a MQTT broker, which is an essential technology owing to the strength of its seamless interoperability to communicate with a broad range of technologies that are operating on a wide variety of platforms. Due to the fact that MQTT has this functionality, IFTTT is able to connect to a broad range of different technologies. The interoperability of IFTTT's many different components is one of the service's strongest selling points. The vast majority of cloud servers provide consumers a difficult method for using IFTTT via the utilization of Webhooks, which ultimately results in a straightforward URL request that requires very little work on the part of the user. IFTTT users have access to a limited number of cloud services, each of which comes with its own user interface that may be accessed directly by the user.

Storage limits

In order to make use of a device that is connected to the Internet of Things (IoT), it is necessary for the user to have access to a sufficient amount of cloud storage space and time that is provided by the cloud. This is because the data can only be stored for a certain amount of time owing to the nature of the storage medium. This is because the data can only be maintained for a given amount of time, which forces the user to download the data before it can be erased and, as a result, causes it to be assessed. The reason for this is that the data can only be stored for a certain amount of time.

Computing done on the cloud, especially for the processing of data

The calculation of data, on the other hand, involves both arithmetical and non-arithmetical phases, each of which is followed by a method that is explicitly described in order to provide information that is meaningful. It is essential for there to be features and capabilities built into the cloud that make it possible for data processing to take place.

Categories of Services

The vast majority of cloud service providers make available several variants of their very own cloud, each of which comes loaded with brand-new features and perks that vary based on the user's credit balance. These variants of the cloud may be accessed by the user. As a sign of appreciation and goodwill from the cloud, the user's resources are raised to greater levels, and they are rewarded with extra credits as a result. These higher levels of resources can be earned by providing financial support to the cloud or by promoting the cloud to other users, which requires the user to complete a survey on the various types of services offered by the cloud. Other ways to earn these higher levels of resources include donating to the cloud and writing reviews of the cloud's services. Donating to the cloud or posting evaluations of the services offered by the cloud are two more methods that may be used to get these greater levels of resources.

Servers

There is a potential that a cloud-based system will be unable to function as a server that is used on its own. The cloud's dependability, redundancy, performance, capacity to offer continuous service, capability to be withdrawn for maintenance, client handling capabilities, and geographic reach out might all be improved by extending the cloud to include numerous servers. One technique for calculating the total number of servers is to use DNS protocols to investigate the IP addresses of the servers. This is only one of many possible approaches.

Performing Calculations on a Number of Different Aspects of Safety

Users of the internet of things have a vital expectation that their personal information and sensitive data will be kept secure at all times. Because of the growing dangers posed by hackers who are striving to get into as many Internet of Things devices as they can, network security should be given the highest priority when building a system. This is because of the increased number of devices that are connected to the internet. These thieves want to get access to as many different devices as they possibly can. As a result, security is particularly crucial since it is necessary to limit the intelligence and subversion skills that hackers possess. This allows him to establish the level of security that is there. In order to carry out this computation, the author of this research visits a great deal of different websites online.

A) Protocol Support

Models of the cloud's protocols are used for the protocol support evaluation, which accounts for 30% of the total weight. The cloud is free to use any of the protocols that are specified (Table 1).

Table I: Protocol Rating

Protocols	Preferenc e	Ratin g
TLS 1.3	1	10.0
TLS 1.2	2	9.5
TLS 1.1	3	9.0
TLS 1.0	4	8.0
SSL 3.0	5	6.0
SSL 2.0	6	4.0

B) Key Exchange

Hackers are able to launch attacks (including MITM and others) against a system that does not entail the exchange of keys during the authentication process. Because these assaults enable the hacker to get access to the communication channel, they eventually work to the hacker's benefit; as a result, the key should be exchanged in a safe way. The key exchange is made up of two distinct processes, and it carries a weight equivalent to thirty percent of the total.

(i) User verification is one of the steps, and it is the obligation of the cloud service provider to guarantee that its subscribers and users are who they claim to be.

(ii) The next step is ensuring the safety of the data transmission itself by protecting the generation, transfer, and exchange of code words. This may be accomplished by ensuring that the steps are carried out in an encrypted format. The second stage is the one that comes after the first stage.

Clouds	Adafruit IO	AWS	Blynk	Thingspeak	Ubidots
Total Samples	336	433	687	577	596
Average Latency	178.67 ms	25619.03 ms	76.34 ms	1255.48 ms	1265.07 ms
Maximum Latency	920 ms	35977 ms	822 ms	2702 ms	2891 ms
Minimum Latency	59 ms	10160 ms	56 ms	849 ms	819 ms
Latency Variation	47%	8.2%	62.4 %	26.4%	24.2%
Packet Loss	12 per 1000	No loss	No loss	No loss	2 per 1000
Throughput	98.81%	100%	100%	100%	99.83%
Interval for update	2 sec (30 data/min)	0 sec (any feed)	0 sec (any feed)	10 sec (32 feeds)	0 sec (any feed)
Min. update time	0 sec	0 sec	0 sec	10 sec	0 sec
User Friendly	YES	YES	YES	ACCEPTABLE	YES
Widget creation	No creation	Allows creation	No creation	Allows creation	Allows creation
IFTTT compatibility	Direct Involvement	Restricts IFTTT	Through Webhooks	Through Webhooks	Through Webhooks
Security Rating	9.65/10	9.15/10	9.45/10	9.45/10	9.45/10
Protocol	TLS 1.2	TLS 1.2	TLS 1.2	TLS 1.2	TLS 1.2
Key Exchange	RSA 3072	RSA 2048	RSA 3072	RSA 3072	RSA 3072
Cipher Bits	SHA 384	SHA 256	SHA 256	SHA256	SHA256
Data Computation	None Available	Multiple Platform	None Available	MATLAB	Own Platform
Storage Space	1.5 GB	5GB	Unlimited	3 million data / year	Unlimited
Storage Time	1 month	No limits	3 months	No limits	2 years
Servers	2 *servers*	1 server	1 server	2 server	2 servers
IP Address Server 1	52.72.201.158	99.84.226.218	185.203.72.17	54.172.155.251	IPV4: 0.23.124.67
IP Address Server 2	2.7.124.212			54.210.243.251	IPV6: 2607:f0d0:2101:94:0:0:0:4

As Equation 1 shows, users of Adafruit IO and Blynk will only encounter delay when the cloud transmits data in response to a change in the monitored variables. Since AWS, Thingspeak, and Ubidots only choose to carry data when it is specifically requested, the latency is calculated using Equation 2. The standard deviation of the Adafruit IO's latency measurements is equivalent to 47% of its average latency, which is a lot considering how low the latency itself is. Amazon Web Services' performance suffers from the delay, and the service's standard deviation of ranges is lowered by 8.2 percentage points as a result. Blynk's latency is great generally but very varied (62.4 percentage points, to be exact). The standard deviation of Thingspeak's latency is 26.4%, which is well within acceptable limits. Once again, Ubidots shows above-average performance, with a latency variation of just 24.2% of the mean. The Adafruit IO has a performance level that is modest, with a packet loss rate of 12 per 1000. Ubidots is reduced to tolerable levels since 2 out of every 1000 packets are lost due to congestion. AWS, Blynk, and Thingspeak, all of which provide 100% throughput, contribute much to its remarkable performance. Adafruit IO limits the update rate of connected IoT devices to 30 updates per minute, independent of the number of feeds being aggregated. With the minimum update time set to zero seconds, the user has access to all 30 settings within the first few seconds of using the app. Incredibly, Thingspeak only allows IoT devices to update once every ten seconds over thirty-two channels. Users of this platform, unlike those of Adafruit IO, are limited in their capacity to make on-the-fly adjustments to the channels, and instead can only do so at regular intervals. With AWS, Blynk, and Ubidots, the minimum update time for IoT devices is always 0 seconds, indicating that users are free to modify the values whenever they choose.

Adafruit IO, Amazon Web Services (AWS), Blynk, and Ubidots all include user-friendly functions including feed management, trigger execution, dashboard visualization, and histogram plotting, as well as public and private sharing, activity tracking, key regeneration, and widget-based input visualization. Public and community sharing, as well as activity monitoring and key regeneration, are additional user-friendly features. Key regeneration, public/community sharing, and activity tracking are some of the other features. While Adafruit IO has several capabilities that aren't available elsewhere, such a weather update, maps, and remote control, it doesn't provide customers the means to create their own widgets. Users may tailor-make their own widgets with the help of these technologies. Amazon Web Services (AWS) is a prime example of the hegemonic nature of the cloud since it supports a large number of platforms that are exclusive to AWS and does not allow the use of any third-party applications. Some of the alternatives that are available from Blynk include joysticks, live video feeds, email clients, audio players, and GPS feeds. Other possibilities include various widgets and gadgets. Downloading the Blynk app, which is compatible with mobile devices running the iOS and Android operating systems, is the only way to get access to the company's services. The clients of Blynk do not currently have access to any web-based products from the company. The accessibility of Ubidots, which may be achieved both via a website and through mobile applications, contributes to the user-friendliness of the platform. It offers support for a ton of additional amazing features, in addition to a ton of other fun features. Even though

Thingspeak only offers a limited selection of widgets, it lays the groundwork for users to create their own widgets, which is a significant benefit. The ability for customers to alter their IFTTT settings from inside the IFTTT interface is a distinguishing feature of Adafruit IO's superior IFTTT service. This capability contrasts well with those offered by competitor clouds. IFTTT is unable to get assistance from AWS, and the company does not allow for indirect interactions to take place via the use of Webhooks. In spite of this, it has support from AWS Lambda as well as Amazon Alexa. Alexa and Lambda are two services that were built by Amazon Web Services (AWS) on its own platform. These services are responsible for handling voice control and home automation, which are two of the most notable characteristics of the cloud. Webhooks make it possible for users of Blynk, Thingspeak, and Ubidots to connect to the IFTTT service. The functioning of the cloud is severely restricted as a result of this.

Results & Discussion

Adafruit IO

Adafruit IO is a fantastic solution for time- and trigger-based applications that demand minimal resources and channels. These applications include those utilized in micro-Internet-of-Things projects for industrial and domestic automation. Adafruit IO is an adaptive cloud that is easy to use and provides a wide range of features, such as minimum latency, quick updates, high-caliber security, and the use of various servers. Adafruit IO also utilizes several servers to increase its capacity. The user-friendly capabilities of the cloud include the ability to manage a variety of feeds, SMS and email publishing tasks, community sharing, activity tracking, trigger events, and the ability to monitor activity. In addition, the cloud is able to publish various types of content, such as emails and SMS messages.

Pros:

Adafruit IO does an excellent job of supporting the creation of sophisticated graphic visualization tools for the purpose of understanding the on-screen motions of the device, which is the primary goal of this endeavour. The cloud provides low update times and low latency. The server's provision of a premier IFTTT service, which increases cloud accessibility, stands out in stark contrast to competing cloud offerings. Cloud services are unique in that they use encrypted protocols to increase user security and privacy. The cloud's availability of redundant servers increases its stability and dependability.

Cons:

Adafruit IO's data refresh rate is only average, it doesn't support synthetic variables, and there's no way to compute with the data. Since the cloud only provides access to a minimal set of data streams, a dashboard, and resources, its uses are constrained to less extensive forms of automation. Since packet loss is so significant, the throughput ratio is a major bottleneck for cloud performance. The cloud has a low latency, but the latency might fluctuate at any time, making it unstable. The user is unable to acquire a new widget of his choosing due to the cloud's restrictions. IoT devices with complicated functions often need a lot of storage space and time, but these resources are insufficient.

B) AWS

Amazon Web Services (AWS) increases the financial value of the Internet of Things by making it simpler to connect any piece of hardware, from sensors to gateways. Amazon Web Services (AWS) was first introduced to the public in 2002. Numerous subscriber actions and call operations are coordinated for the purpose of ensuring everyone's safety. A wide range of encryption keys may be used in conjunction with AWS to ensure the safety of all data that is stored, all devices that are linked to it, and the many different types of network topologies. In addition to the typical AWS key ID and secret key, you will need a region, endpoint, and domain that are associated with AWS IoT. Because of the use of powerful encryption, each of these choices offers the highest possible level of safety. The server is well-suited for broader implementation in industrial and home automation because to its greater capacity to filter, correlate, and aggregate real-time data streams. This ability makes the server particularly useful. These apps make a very good use of the resources that are available on the server.

Pros:

When it comes to computing data, AWS shines, making it easier for the user to conduct an in-depth analysis across many native platforms. If you have several Internet of Things devices, you can be certain that they will all be able to communicate with one another without any hitches thanks to AWS's unrestricted cloud infrastructure. The cloud has very little latency fluctuation. If you need to send sensitive data over AWS, you can be certain that it will arrive safely and quickly thanks to the cloud's high throughput. Voice-based apps are made easier with the inclusion of Alexa-enabled devices.

Cons:

AWS's limited functionality is due to its extraordinarily high latency. AWS has dominant tendencies that don't let third-party apps to alter its goals or workflow. Using standard protocols and encryptions might mess with the cloud's security settings. Since AWS doesn't allow for integration with IFTTT, non-AWS apps are constrained.

C) Blynk

For mobile data visualization, the popular cloud service Blynk is an excellent option. Blynk is able to collect data via its servers, make it accessible for streaming on mobile platforms, analyse and display massive amounts of incoming data series in real time as it hits cloud servers, and more. The drag-and-drop functionality of Blynk becomes even more responsive to the users' particular requirements when the users include live charts and grids with their preferred forms. Event monitoring, location tracking, and mobile apps that are easily maintained via user-defined email and text messages may be of tremendous utility to large-scale non-computational industrial and home automation systems.

Pros:

Blynk's benefits include its compatibility with a wide variety of real-time use cases, the high quality of the data visualization it provides, its low latency and app-based operation, its reasonable one-time price for feeds, and its exceptional average latency, update time, and throughput. In addition, Blynk's app-based operation and low latency make it ideal for use with mobile devices. The fact that cloud storage offers customers with a time period of three months during which they may save their data is particularly tempting since it reassures them that they will have plenty of time to retrieve their data. The one-of-a-kind widgets that Blynk provides, which include a music player, a Twitter feed, a joystick, an accelerometer, a GPS trigger, an SMS notification, and an email trigger, are quite helpful.

Cons:

Blynk is unable to do data calculations and does not permit the use of synthetic variables since it is a cloud that is built on a single-server app and is not supported by browsers. The cloud suffers from inconsistent latency and does not enable users to create their own widgets; if this then that is only enabled in part via web hooks.

Conclusion

The Internet of Things is proving to be a highly valuable tool in the automation of both industrial and domestic settings, where it has a large potential for applications and several potential benefits that it may provide. This is because the Internet of Things has the potential to provide all of these benefits, and more. This is due to the fact that the Internet of Things has the capability of delivering all of these advantages, as well as many more. In order to investigate the various aspects of the cloud that are required for a high-quality Internet of Things interface, we investigated five of the most well-known cloud-based platforms. Adafruit IO, AWS, Blynk, Thingspeak, and Ubidots were the platforms used. It was our intention to identify the different facets of the cloud required for a top-notch user experience. These are only a few of the different platforms that we investigated. The automation must be installed perfectly in homes and businesses to ensure that all data transfer for related operations is carried out at an extremely high level. Most applications need specialized hardware and software in order to perform functions like controlling, data uploading/monitoring/calculating, voice controlling, platform switching, trigger or time based application, email/messaging, and even block chain. Controlling, data uploading, monitoring, and calculating; voice controlling; platform switching are some of the activities that come under this category. Processes such as controlling, uploading data, monitoring, and calculating, controlling with one's voice, and switching platforms come under this heading. The user may do an examination of the cloud's properties using the methods provided in this article, and then choose the cloud that is best suitable for their specific purpose and needs if they desire an application that does not fit into these categories. This paper outlines the procedures that can be followed to conduct an analysis of the features of the cloud. This article provides an overview of the processes that may be implemented in order to carry out a study of the characteristics of the cloud. After finishing the research, we got to the conclusion that the open source cloud has to have an update implemented so that it may support the following parameter value. Because of this discovery, we believe that it would be beneficial to make this adjustment. By offering the most pleasant experience that can be provided, this will ensure that customers have the greatest possible experience when utilizing the cloud platform.

References:

1. Abdur Rahim Biswas, Raffaele Hiaffreda: IoT and Cloud Convergence: Opportunities and challenges in IEEE world forum on internet of things (WF-IoT), pp. 375 - 376, 24 April 2014.
2. Partha Pratimray; A survey of IoT cloud platforms in Future computing and informatics journal, pp. 35-46, December 2016.
3. Luciano Barreto, Antonio Celesti, Massimo Villari, Maria Fazio, Antonio Puliafito in An Authentication model for IoT clouds, IEEE/ACM International conference on Advances in Social networks Analysis and Mining (ASONAM), 11 February 2016.
4. Ran Canetti, Ben Riva, Guy N. Rothblum, Practical Delegation of Computation using Multiple servers, proceedings of the 18th ACM conference on Computer and Communications Security, pp. 445-454, 21 October, 2011.
5. Boyi Xu, Lihong Jiang, Athanasios V, Vasilakos, IoT-Based Big Data Storage Systems in Cloud Computing: Perspectives and Challenges in IEEE Internet of Things Journal, pp. 75 - 87, 20 October 2016.
6. Nirandika Wanigasekara, Jenny Schmalfuss Darren Carlson, David S. Rosenblum, A bandit approach for intelligent IoT service composition across Heterogeneous smart spaces in proceedings of the 6th International conference on IoT, pp. 121-129, 7 November 2016.
7. Yuang Chen, Thomas Kunz; performance evaluation of IoT Protocols under a Constrained Wireless Access Network in International conference on selected topics in Mobile & Wireless networking (MOWNET), pp. 1 – 7, 23 June 2016.
8. Davide Mulfari, Antonio Celesti, Massimo Villari, A Computer System Architecture providing a User-Friendly Man machine interface for accessing Assistive Technology in Cloud computing in Journal of Systems and Software, pp. 129-138, February 2015.
9. Steven Ovadia, Automate the Internet with "If This Then That" (IFTTT) in Behavioral & Social Sciences Librarian, vol. 33, 2014, pp. 208-211, 10 November 2014.
10. Qi Zhang, Lu Cheng, Raouf Boutaba, Cloud computing: state-of-the-art and research challenges, in Journal of Internet Services and Applications, pp. 7–18, 20 April 2010.
11. Zia ur Rehman ; Farookh K. Hussain ; Omar K. Hussain, Towards Multi-criteria Cloud Service Selection in 2011 Fifth International Conference on Innovative Mobile and Internet Services, Ubiquitous Computing, pp. 44 – 48, 4 August 2011.
12. SAVIO SCIANCALEPORE, ANGELO CAPOSSELE, GIUSEPPE PIRO, GENNARO BOGGIA, GIUSEPPE BIANCHI, KEY MANAGEMENT PROTOCOL WITH IMPLICIT CERTIFICATES FOR IOT SYSTEMS IN IOT-SYS '15 PROCEEDINGS OF THE 2015 WORKSHOP ON IOT CHALLENGES IN MOBILE AND INDUSTRIAL SYSTEMS, PP. 37-42, MAY 18, 2015;
13. Yosra Ben Saied, Alexis Olivereau; HIP Tiny Exchange (TEX): A distributed key exchange scheme for HIP-based Internet of Things in Third International Conference on Communications and Networking, pp. 1 – 8, April 1, 2012.
14. Mouza Bani Shemaili, Chan Yeob Yeun, Khalid Mubarak, Mohamed Jamal Zemerly, A new lightweight hybrid cryptographic algorithm for the IoT 2012, International Conference for Internet Technology & Secured Transactions, pp. 87 – 92, 12 December 2012.

15. Sachin Babar, Parikshit Mahalle, Antonietta Stango, Neeli Prasad, Ramjee Prasad, Recent Trends in Network Security and Applications in International Conference on Network Security and Applications, CNSA 2010, pp. 420-429, 17 November 2016