Software, Systems and Applications - Cloud Computing

hzwr87

For this scenario the product we will consider is an internet enabled car. This would record data about how the various components of the car are functioning under various conditions, while also monitoring the mileage covered by the user and analysing their driving style, in order to provide information to the manufacturer as to why some components are easily damaged, perhaps when used in certain conditions that the manufacturer may not have been able to produce in testing. In this document, we shall examine the considerations this company should make for implementing an IT infrastructure for this purpose, and recommend a system for them to use.

This system will require involve each car sending condition updates and data regarding the driving style of the owner back to the manufacturer at regular intervals. This means that the manufacturer will require a server of some kind to handle this communication, and a large database to store the data sent back . Once this data has been received, it must then be processed effectively using a machine learning algorithm to provide the company with effective statistics for the condition of their cars in different real world circumstances. The servers and databases should be handled by Infrastructure as a Service (IaaS), and the processing of the data could be handled by Software as a Service (SaaS) if the selected cloud provider has software that would fill the requirements, it the provider does not have appropriate software available then this requirement could be filled by purchasing Platform as a Service (PaaS) from the provider and implementing the company's own analysis software on top of this.

In this scenario, pushing the majority of this functionality to the public cloud has many advantages. The first of these is mass storage for the database component of the system, as this can be considered a Big Data system. For the car company we are discussing here, this would be very important, as they would have a huge amount of cars on the road all sending back a large amount of data on the status of the car and how it has been driven. If the company were to purchase sufficient hardware to perform the functionality they require, it would cost them a huge amount. As a consequence of this, it would be better to utilise the hardware of a cloud service provider, as this functionality, while useful to the company, it not business critical.

Also, the amount of hardware required for this purpose is variable. Once the first model of car with this capability on board is released, for the first few weeks and months the number of cars with this technology in them will be relatively few, so the number of servers and databases that will be required will also be relatively few. As more of the cars are sold, more data will be sent and received, and thus more hardware will be required. The amount of communication will also vary at different times of the day, being much higher at rush hour when people are commuting, and at the weekend when people are travelling. The cloud approach will mean that the amount of hardware needed will be able to vary to respond to demand, which is a far more efficient use of money than the company buying their own hardware and having it always running irrespective of demand.

Another advantage of using a cloud provider to provide this functionality is the availability of data. A company like the one we are considering here is likely to have many offices all around the world, and if they handled this in house, it would have to be either all at one location or split into smaller chunks around multiple locations. This could lead to a problem of lack of availability of data to some locations, which would be mitigated by using a cloud system. Cloud providers also have servers based all around the world, so for our car company, that would operate globally, this distribution of servers would streamline the process of gathering data from many locations.

One of the major focuses of the major cloud providers currently is security. As a result of this, they are investing large amounts of their resources in developing their security capabilities, and as such are considered a safe bet to provide the functionality that many companies require. Developing effective security requires a tremendous investment, and given that this investment has been made by the public cloud providers, this means that companies like our car company do not have to incur these costs themselves. This is another clear advantage of using the public cloud.

In this situation, security is very important as some personal data is being recorded. Even if the system does not directly store information about who the owners of the cars are, it will have information such as registration plates or internal company car IDs that could be used to map cars to owners. As the data being recorded is based on positional data, if the aforementioned association between the cars and people can be made then the data can be used to track people, which leads to legal and ethical problems. It is therefore very important for the car company that they ensure that this data is secured appropriately, both by them and their choice of cloud provider, as the customer is responsible for security in the cloud, whereas the provider is responsible for the cloud [20]. They must ensure that the provider that they select has an appropriate data protection policy in place, with particular emphasis on the data only being made available to the company, or those that they have approved.

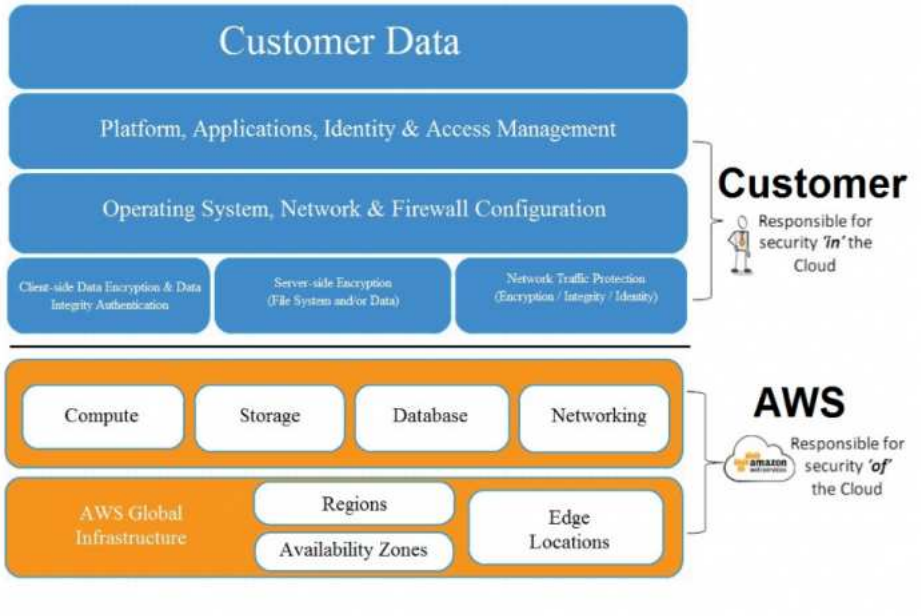

Figure 1: Security Responsibilities According to Amazon[20]

Despite these advantages, there are some critical points of failure associated with a public cloud system. When purchasing a cloud solution, the customer must have trust in the provider to

give them appropriate performance, availability and storage capacity as agreed upon in a the contract between our car company and the provider. In this contract, there must be clear stipulation of what should be and will be done if any of the cloud components fail. One of the critical points of failure will always be one of the components failing for a variety of reasons, from powercuts to failure in the components themselves. This failure should be mitigated by the provider, as for some products the functionality the failed service was providing is switched to another piece of hardware. This failure will not end up being client facing unless the service being used does not support being transferred across to other hardware. As most mainstream providers have an uptime of close to 99.9% however, this is unlikely to cause a huge problem given that thus functionality is not business critical.

Another potential point of failure, further up the stack, is the data processing software, if a SaaS product is being considered. Any problems with this would normally be thought to be the responsibility of the customer rather than the provider, but within certain cloud infrastructures, many machine learning algorithms are included as part of their software offerings. Whilst in the majority of cases problems will indeed have been caused by the customer, the potential for failure on the side of the provider does still exist, and should be mitigated by extensive testing. If such an error was to occur, it could cause incorrect results, inefficient use of resources and even data corruption. These consequences would definitely be client facing as they would potentially recieve incorrect data, leading them to make changes to their cars that may not actually be needed.

When selecting a cloud provider, there are many cost considerations that should be taken into account. The first of these is whether they charge for use over a period of time, or just for usage. For instance, in our example, if all the cars that are currently in operation transmit their information every hour, then the servers are only required for this short time slot rather than being operational all of the time. Therefore, in the scenario that we are considering here, a provider that can accommodate this demand in their pricing scheme would be preferable.

In addition to this, it would also be preferable for our company to be charged only based on the amount of data that is being transmitted at any one time. As mentioned previously, if the data is only transmitted for active cars every hour, then during busy times of the day when people are commuting to and from work or at weekends, there will be far more active cars than at other times of the day. Therefore, as before it would be preferable for the company to only be charged for as much traffic and storage as they need at certain times of the day.

Similar cost considerations should be taken into account for the virtual machines that perform the processing of the data. Such processing is only necessary when the analysts for the company wish to examine the data and draw conclusions from this. As a result of this, the company will only want to pay for the amount of time that their analysis software is actually running.

Assuming that a provider can be found that satisfies the above conditions, it will be fairly simple to predict the costs. As the company will know how many cars they have sold, they can make a fair estimate based on national driving statistics of what proportion of their cars will be on the road at certain times, and therefore how many cloud resources they will need at different times of the day. They will also have set up a schedule for when they will analyse the data, and this can be adjusted to find the balance between cost of the cloud resources required to analyse the data, and the frequency of analysis required to get meaningful results.

Taking all these considerations into account, we must now select a cloud provider that best fits the needs of the company. The providers that we will consider here are Amazon Web Services

(AWS) and IBM Bluemix, as these are two of the market leading providers at the time of writing, as can be seen in the figure below.

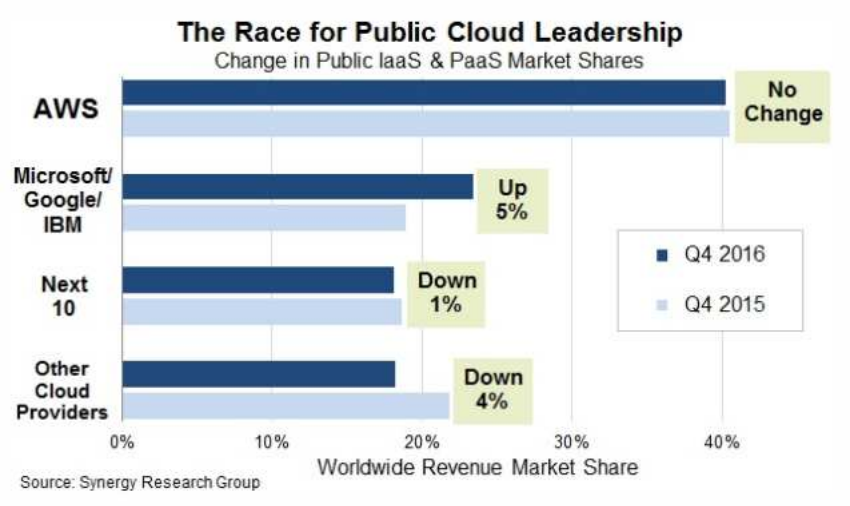

Figure 2: Current Cloud Market Share [12]

We will first consider what levels of security each provider promises. Amazon considers security their "highest priority" [16], and claim that all of their datacentres are secured with strong safeguards in place, and no matter the amount of data that a client needs to store, they will benefit from the full secure package that AWS provides. Amazon has complete commitment to securing the data of their customers, and will never disclose it unless required to by a legally valid and binding order. IBM provide similar guarantees for their Bluemix system, stating that all Bluemix systems adhere to "security policies that are driven by best practices in IBM for systems, networking, and secure engineering". [2] More detail on how this is ensured is provided in the figure below. With both of these providers however, once the data has been brought out of the cloud and on to local systems, then it is completely the responsibility of the customer to ensure that the data continues to be secured.

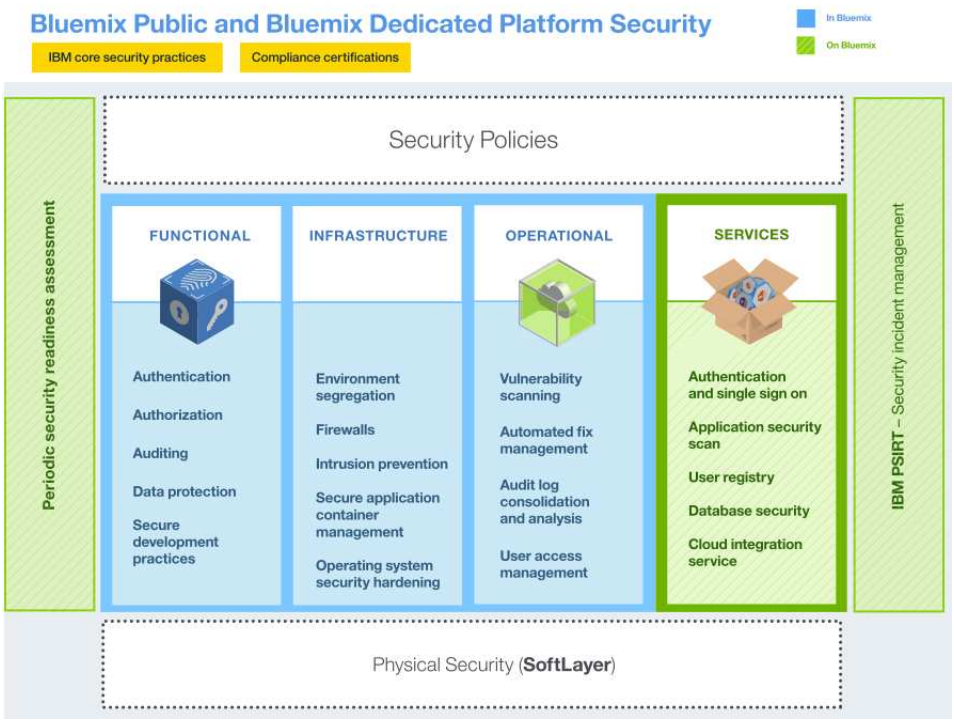

Figure 3: Bluemix Security [2]

Next, we will consider the uptime guarantees of each provider. AWS guarantees an uptime of 99.9% in their Service Level Agreement (SLA). However, over the last 30 days, the CloudHarmony comparison website [7] states that the uptime of AWS services was 99.9708%. IBM guarantees an uptime of 99.95% in their SLA for instances configured for high availability and are spread across Bluemix regions over the course of a month. Downtime is defined here as the total minutes where Client is unable to connect to at least one of two instances. However, as mentioned before, the CloudHarmony comparison website [7] states that the uptime of Bluemix services was 99.7721%. As we can clearly see, both of the cloud providers promise very high uptimes, and although IBM have not quite achieved their promise in the last month, the uptime is still impressive, especially as the functionality being considered here is not business critical, and the cars will be unaffected by outages, they will just not be able to send any data back.

Finally, we will consider how the IaaS, PaaS and SaaS offerings differ. In IaaS, Amazon is by far the market leader in this area, with their EC2 service. This service supports the flexible payment plans discussed above that would be desirable in this scenario. Spot instances would also be appropriate, as this is not business critical functionality, the company could specify how much they would be willing to pay for data collection, and then only get it when hardware is available for this price [13]. This would be a good way of getting an appropriate amount of data for the price that the company is willing to pay, rather than what Amazon is charging. IBM's counterpart to this is their virtual servers system [3], which functions in a fairly similar way, but does not support spot instances payment. This makes it difficult to compare the price, as with spot instance AWS should be cheaper, but this greatly depends on demand. Even so, without considering spot instances, EC2 is cheaper for the vast majority of configurations. This currently seems to be the best choice to handle the server requirements.

To staisfy the Big Data storage needs of the customer, AWS also provides the Simple Storage Service (S3) [15]. This will provide a cheap and flexible interface to store and retrieve large amounts of data, which will be ideal for storing the data which will be sent back by the large

amount of cars that this company will have on the road. IBM's alternative to this is their Object Storage Service. Both of these systems appear to use the same storage protocol, that of Amazon S3, so the choice here should be made to align with the rest of the system. [6]

Amazon Lambda is a good PaaS offering, built on top of EC2, that abstracts away all of the virtual machine (VM) components, as it just requires code that needs to be run, and the Lambda platform takes care of the rest. This is a good service to use because it enables migration of services in case of failure and avoids the company having to deal with provisioning, configuration, monitoring, managing, and maintaining VMs that they use.[18] IBM have a comparable PaaS offering to AWS Lambda, called OpenWhisk, that is also built on top of their virtual servers, and abstracts away the lower level functionality to speed up and simplify use for customers. It also offers easy access and integration with IBMs SaaS offerings such as Watson.[4] In this scenario these services would be useful to handle the communications coming from the cars and passing the data on to a database to be stored and analysed, as well as for potentially performing the analysis if an appropriate SaaS offering is not available.

Both of the providers that we are considering here have an extensive range of SaaS offerings in their marketplace, built to satisfy a wide variety of requirements. For our scenario, we are mainly interested in the machine learning functionality that SaaS can provide to enable the company to effectively process the data that their cars are sending back, and trying to map driving style to the condition of the car compenents is a problem that can be solved using Machine Learning. The AWS marketplace has a wide variety of machine learning based offerings, partnered with 47Lining and NorthBay, to solve a wide range of machine learning related problems [14]. For IBM, one of their major selling points in terms of cloud services can be found in their SaaS, in the form of the Watson API, which brings state-of-the-art cognitive technologies to the cloud to solve a large range of problems[1]. However, after searching through the marketplaces and documentation, neither of them appears to have a SaaS offering that would fully fill the data processing needs of our car company.

In Conclusion, as both technologies develop we can observe that the internet of things and cloud computing are naturally joined together. We believe that the use of a public cloud solution for the scenario being discussed here would be highly advantageous. Therefore, following the analysis of the offerings provided, the solution that we are suggesting is to use an Amazon system to solve this problem. As they are the market leader in cloud services, and have been for many years, they can afford to have their pricing set cheaper than their competitors and invest huge amounts of money into research and development, improving their own system. The economies of scale caused by this mean that many of their services remian the best, for the cheapest price. With regards to IBM, the need for Watson is key for whether their service is the preferable option or not, as this is the best offering they bring to the market. Their other services are still good, and provide similar functionality to Amazon, but are more expensive. In terms of components for the solution, we recommend using S3 for the Big Data requirements, EC2 spot instances for the servers and potentially Lambda for controlling the servers at a higher level, as well as providing a platform that the company's own data analysis system can be implemented on top of.

References

[1] IBM Bluemix. Unlock the power of cognitive with watson, 2017.

[2] IBM Bluemix. Bluemix security, January 2017.

[3] IBM Bluemix. Ibm bluemix openwhisk, January 2017.

[4] IBM Bluemix. Ibm bluemix openwhisk, January 2017.

[5] Rachel Burnett. Cloud computing and data protection, February 2011.

[6] IBM Knowledge Centre. What's new as of march 2017?, March 2017.

[7] CloudSquare. Research and compare cloud providers and services, 2017.

[8] BBVA Communications. Eight reasons to use cloud computing in the company, September 2016.

[9] Pamela K. Isom and Deborah Hawley. Strengthening your business case for using cloud, July 2010.

[10] Shubham Mittal. Eliminating single points of failures on aws cloud, April 2016.

[11] Information Commisioner's Office. Guidance on the use of cloud computing, 2012.

[12] Joe Panettieri. Cloud market share 2017: Amazon aws, microsoft azure, ibm, google, Fedruary 2017.

[13] Amazon Web Services. Amazon elastic compute cloud, 2017.

[14] Amazon Web Services. Amazon machine learning, 2017.

[15] Amazon Web Services. Amazon s3, 2017.

[16] Amazon Web Services. Aws cloud security, 2017.

[17] Amazon Web Services. Aws information requests, 2017.

[18] Amazon Web Services. Aws lambda, 2017.

[19] Mark Skilton and Penelope Gordon. Cloud buyers' decision tree, July 2010.

[20] Peter Tsai. Staying on top of your infrastructure-as-a-service security responsibilities, July 2016.