Cloud Service Models - IaaS, PaaS, and SaaS

Capstone Project MINT 709

By

Afsaneh Rostami

A project submitted in partial fulfillment of the requirement of the degree of

Master of Science in Internetworking

Department of Electrical and Computer Engineering
University of Alberta

Abstract

Cloud includes so many developments and possibilities. Indeed, today's technologies such as high bandwidth networks, virtualization, web interactivity, time-sharing and browser interface have come together in cloud computing. Through the cloud, we would be able to do tasks via a combination of software and services over the internet. Actually, cloud computing helps remarkably to businesses by moving to cloud solutions in order to reduce the costs of purchasing and maintenance servers, IT staff and also providing the possibilities to increase their resources when they need and drop when their peak time passed. Servers in the cloud could be a physical or virtual machine. These days other resources such as network devices, storage, firewall and other security devices have been provided by cloud providers.

We have three types of service which have been provided by cloud providers, Software as service (SAAS), Platform as service (PAAS) and infrastructure as service (IAAS). So many cloud providers exist these days and provide some or all these kinds of services, AWS, Azure, and Google are three top of cloud providers in the world.

Acknowledgment

First of all, I want to thank God Almighty for giving me the strength, the knowledge, the ability and the opportunity to undertake and complete this project successfully. This would not have been possible without his blessings.

In secondly, I want to thank you, my parents, because of their always and warm support during my study even from long distance.

I'm delighted to thank my mentor **Mr. Juned Noonari**, who always supported and guided me and gave me valuable inspiration and suggestions in my search for knowledge. He gave me the freedom to study my project while still making sure that I stay on course and do not deviate from the core of my subject. This thesis would not have been possible without his knowing guidance.
I would also like to express my sincere gratitude to my program coordinator **Mr. Shahnawaz Mir** for providing me such a great opportunity and by allowing me to opt for a project of my own interest.
I am proud to acknowledge my gratitude to **Dr. Mike McGregor**, who allowed me to start this project.
I would also like to express my sincere thanks to all the instructors, professors, seniors, classmates, colleagues and the entire University of Alberta who helped me directly or indirectly in this study and have been helpful and cooperative in giving their support at all times to help me achieve my goal.

Content Table

Figure Table

1.1 Definition

Every organization has a different definition of cloud computing. Wikipedia knows cloud computing as a large group of several servers that networked to each other in order to centralize data storage and online access to computer services or resources. (Shah & Sarkar, Learning AWS, 2015)While the National Institute of Standards and Technology (NIST) defines in another way, cloud computing is a model that everyone can have access to a pool of resources in which they could be able to provision rapidly or release with minimal management effort or service provider interaction. (Shah & Sarkar, Learning AWS, 2015)

Today, the term cloud computing describes the abstraction of the provisional resources which should be accessible over the network. Also, users should be able to add or remove resources on demand and would be charged based on the type and quantum of resources that they consume. Often these cloud-based resources are viewed as virtual. Through their virtual nature, cloud-based resources can be scaled up or down automatically, depending on the load. (Shah & Sarkar, Learning AWS, 2015).

Could computing has some advantages and disadvantages that have been explained in the following.

1.1.1 Advantages:

Following are the well-known advantages regarding using cloud computing have been listed;

- All resources and storage would be available and there is no need to plan in advance and the procurement process.
- It would be possible to use the resources based on the needs. As an example, starting small, and then increasing resources.
- We could be able to make a test and development environment on a smaller scale than the production environment even though we would be able to do during business hours in order to reduce cost.
- We would be able to make an environment the same exact production by using the replica feature in order to improve defect resolution.
- Vertical and horizontal scaling would be so much easy, in order to better manage loads in demand and variations due to business cycles or time-of-day reasons.
- Trying new ideas and software would be possible only by clicking on the needful resources rather than making a plan for resources through the time-consuming process.

- There would be no hardware maintenance or data center operations, migration, and upgrade downtown would be zero. Ease of implementation of high-availability and disaster recovery strategies
- By storing data on the cloud, the failure aspects of storing data to be disappeared. Also, it has another advantage that we would be able to download our data with high speed.
- It would be useful for analyzing large data.
- Through the cloud, we would be able to access some high-end network devices such as firewalls, load balancing, certificate management, and so on very easily.
- The cost of TCO (total cost of ownership) would be reduced remarkably since we will not have to pay for purchasing hardware, hiring some staff to design our network and maintenance it and also redundancy for our data center would be removed automatically.

1.2 Cloud Deployment Models

A cloud deployment model identified how resources would be shared through the cloud. There are three categories of deployment methods.

- Public: third-party service and resources would be accessible for end-users through the internet. The customers' applications and data would be deployed on infrastructure owned and secured by the service provider. Actually, that would be a multitenant environment in which the end-users pay only for the usage of resources. The end-users would not have any accessibility to the location of the data center. An abstraction layer is built on top of the physical hardware and exposed as APIs to the end-user, who leverages these APIs to create virtual compute resources that run in a large pool of resources shared.
- Private: In this model, it consists of all benefits of a public cloud but the service and data would be managed by the organization only for a specific use. Indeed, it would not be a multitenant environment. Usually, private cloud places increase administrative overheads on the customer but give greater control over the infrastructure and reduce security-related concerns. The infrastructure may be located on or off the organization's premises.
- Hybrid: It is a combination of both a private and a public cloud. Some of the service could run on public while some others would run on private. The decision on what runs on the private versus the public cloud is usually based on several factors, including

business criticality of the application, the sensitivity of the data, industry certifications and standards required, regulations, and many more.

1.2.1 Advantages:

In the following, some benefits of public cloud services are explained;

- Utility pricing**:** The end-user would be charged only for the resources it consumes. This provides an opportunity that the end-user would be able to turn on more cloud services when it needs and turn off when there is no need.
- Elasticity: By moving to the cloud solution, end-users would have access to an endless pool of service in which they would be able to increase or decrease the amount of compute resources in order to overcome high loads. While in legacy style, the end-users would have to purchase or lease the necessary resources in order to handle the high load, but this solution was not in real time and it takes time
- Core competency**:** By using the public cloud, the overhead of management would be removed from end-users and more time would be consumed on core competency**.**

1.2.2 Disadvantages:

Below you could find some disadvantages regarding using the public cloud;

- Control: Because end-users do not have any access to data center, so he has to rely on the public cloud vendor to meet their SLAs for performance and uptime. If a public cloud provider has an outage and the end-user has not any plan B, so there would not be any choice for users.
- Regulatory issues**:** Regulations like PCI DSS (Payment Card Industry Data Security Standard), HIPAA (Health Information Portability and Accountability Act), and data privacy issues can make it as a challenge to deploy in a public cloud. It often requires a hybrid solution to meet these regulations.
- Limited configurations: Public cloud Public services have a standard set of infrastructure configurations that meet the needs of the general public need. Sometimes, very specific hardware is required to solve intensive computational problems. In this case, a public cloud would not be a good option.

1.3 Cloud Services Model:

A cloud can interact with a user or application in several ways, indeed through several services. Each cloud service model provides a level of abstraction that reduces the efforts for building a system or even a software.

1.3.1 SAAS (Software as a service)

With this service, customers will use applications from a supplier which is running on the cloud. Actually, customers will not be able to change or modify applications a lot just some minor changes such as placing of corporate logos, language set up and look-feel option. In this case, all layers are outsourced. These applications would be accessible through a simple interface such as a web browser for everybody, from any place at any time. In this report, we will go through Google cloud as one of the known SAAS cloud services and explain some of the main concepts. (Blokland, Mengerink, & Pol, 2013)

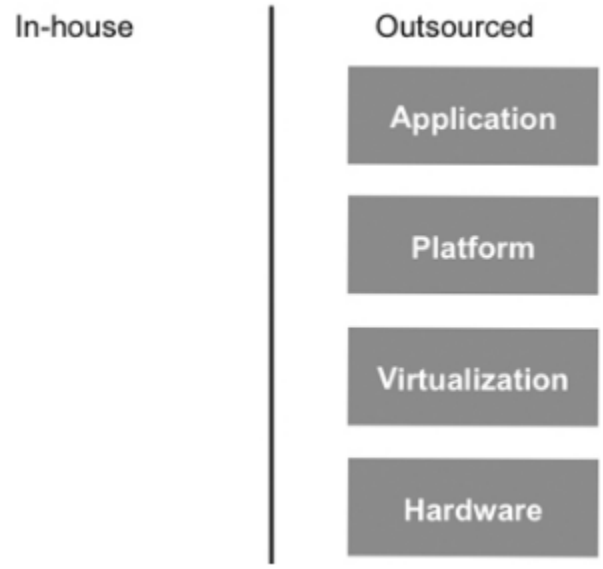

Figure 1. 1

1.3.2 PaaS (Platform as a service)

Through this cloud service, customers would be able to use the programming language and tools that are supported. The customer will not have any control or management over infrastructure but would able to manage the application and some of the configuration of the platform environment. Web hosting would be a known example of PaaS, the companies that provide web hosting actually provide an environment with a programming language such as PHP. In this model, the customer will not be responsible for the platform. In this report will study deeply

about Microsoft Azure, which is one of the well-known cloud services in PaaS. (Blokland, Mengerink, & Pol, 2013)

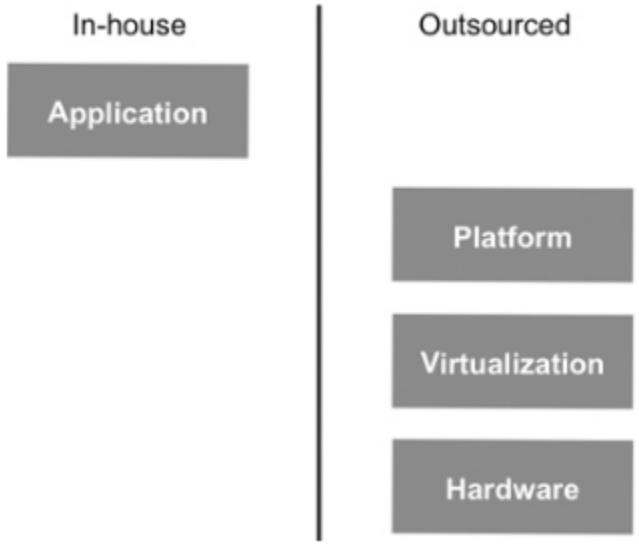

Figure 1. 2

1.3.3 IaaS (Infrastructure as service)

This cloud service provides a virtual data center that developers need to install their own operating systems, manage their database and support software. Actually, the capabilities which are provided to the consumer are accessing to infinite pool resources such as storage, network and other fundamental computing resources where the consumer is able to deploy and run arbitrary software, which can include operating systems and applications. Customers cannot have access to underlying cloud infrastructure in this cloud service but would be able to manage operating systems, storage and deployed applications and control over networking components. Within using IaaS, many of the tasks which are related to managing and maintaining are abstracted and available as a collection of services that can be accessed and automated from code- and/or web-based management consoles. In this cloud service, developers still need to design, and code entire applications and administrators still need to install, manage, and third-party patch solutions, but there is no physical infrastructure to manage anymore. With IaaS, the virtual infrastructure is available on-demand and can be up and running in minutes by calling an application programming interface (API) or launching from a web-based management console.

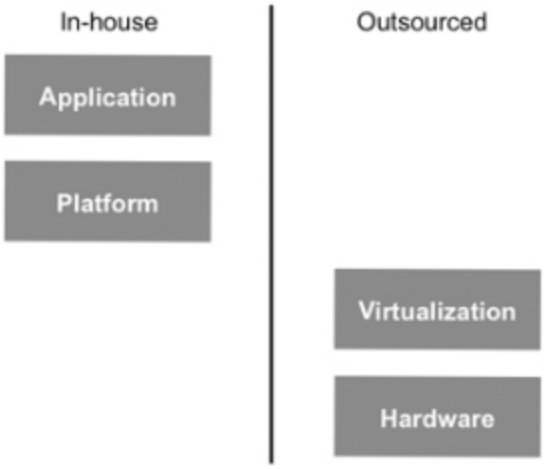

Figure 1. 3

1.3.3.1 Advantages of IaaS:

Advantages of using an IaaS solution include the following:

- The cost of staff for maintaining the data center would be reduced
- If we need to increase hardware, that would be done easily.
- Everything would be virtualized, so we do not need to purchase any hardware.
- We would be charged based on consumption.
- The number of IT staff would be reduced.
- We could have a test environment as well.

1.3.3.2 Server Type:

Within IaaS cloud service, there are several types of servers that customers could choose their desire;

- Physical server: This type of server is not shared among two or more customers and is more expensive.
- Dedicated virtual server: With installing Hypervisor on one server, we could be able to run multiple OS, which there is no need to be same. Each operating system is protected from others on the server and could be configured by the customer. The virtual server is used by only one customer, which, again, will result in a high cost.
- Shared virtual server: In this type of server, one virtual server would be shared among several customers. For example, provide web server capabilities to multiple users. The customer cannot configure the shared virtual server.

1.4 Why not a cloud

There are situations that the cloud is not the best option even though it could save cost. For instance, Google's infrastructure is a big target for attacks such as denial-of-service attacks, and government espionage and has the budget and expertise to build gigantic infrastructural footprints. In the following trend, it shows the usage and cost pattern. Actually, it shows that companies have to spend more cost to keep available resources over than needed resources. (Geewax, 2018)

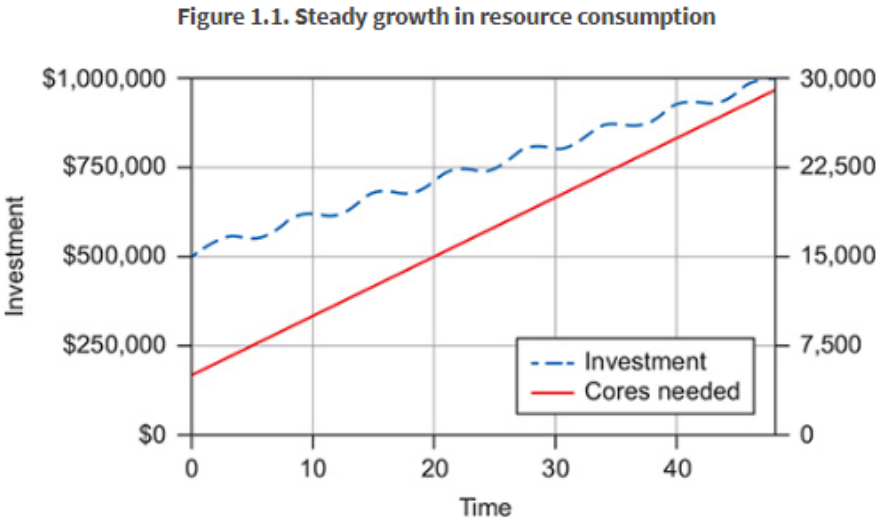

Figure 1. 4

Also, another chart has been shown in the following that illustrates a more typical company of internet age where growth is increased suddenly and dropped without any notice. In this case, company had to provide resources for top usage while it would be excess at other times. (Geewax, 2018)

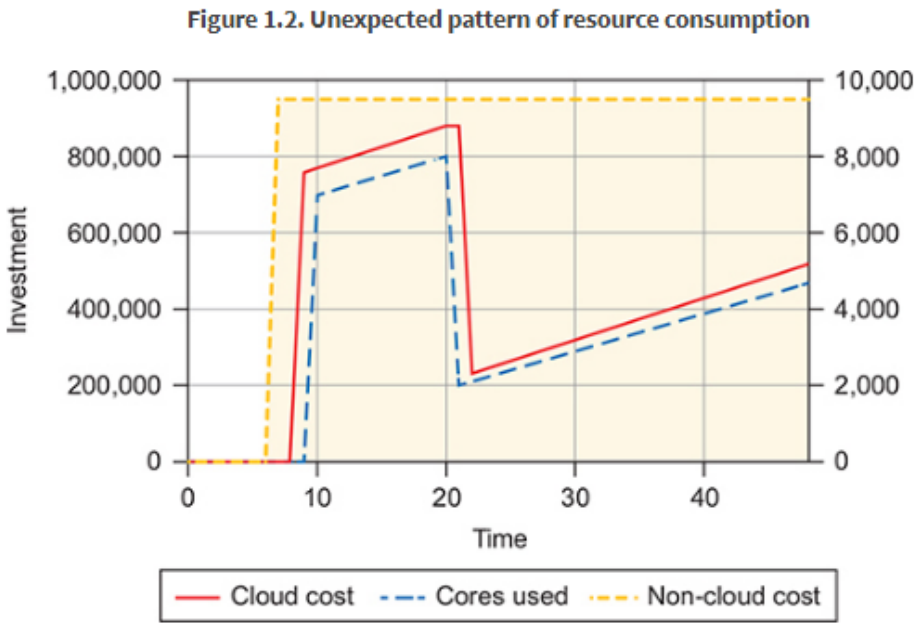

Figure 1. 5

In summary, if there is some expertise in the data center of a company who could identify the needs for the current time and have plans for future in order to provide needful resources synchronous with the growth of the company, a cloud would not be a good option but if we do not know what we need exactly for today and for future, the cloud would be a good option. (Geewax, 2018)

2. Google cloud

In his report, we will study Google cloud applications which have been known as one of the famous SaaS providers in the cloud service concept

2.1 Cloud application

From many aspects, an application for a cloud would be same as an application for local servers. But there is a difference, when we are writing a program, we need to deploy thing such as binaries on particular servers and think which servers are handling which things. While cloud application relies on hosted services whenever possible. For instance, consider an application that we could upload our data through it. In a traditional way, we stored our data in relational database and whenever someone wanted to see the content, it would be retrieved from the database and return it through the web server, the same as the below diagram.
(Geewax, 2018)

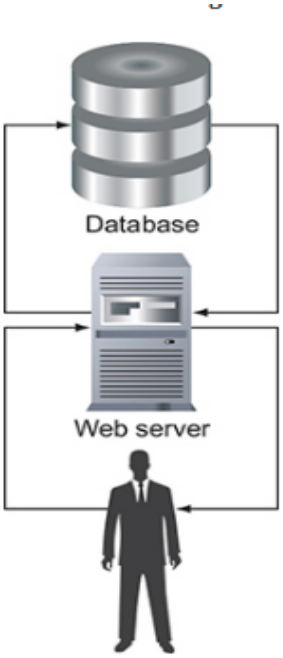

Figure 2. 1

This way has some disadvantages. First of all, it has some storing binary data in the database, which is inefficient, and actually we are putting extra load on the database. For this case, we could save photos somewhere on disks and then use the static serving capabilities of webserver to deliver photos. In this case, database would be free to work on more important things.

For this design, we have a web server, which is located in a place, so if a person from far distance requests for a photo of that web server, all data should travel in order to respond the request. So, for this problem, we could use edge cashing or using CDN. By using the cloud, it provides managed services that solve problems from 1 to 100. In which, when users upload a photo to the web-server, we could resize it and edit it, and then forward the final image along to Google Cloud Storage, using its API client to ship the bytes securely, the same as the below diagram; (Geewax, 2018)

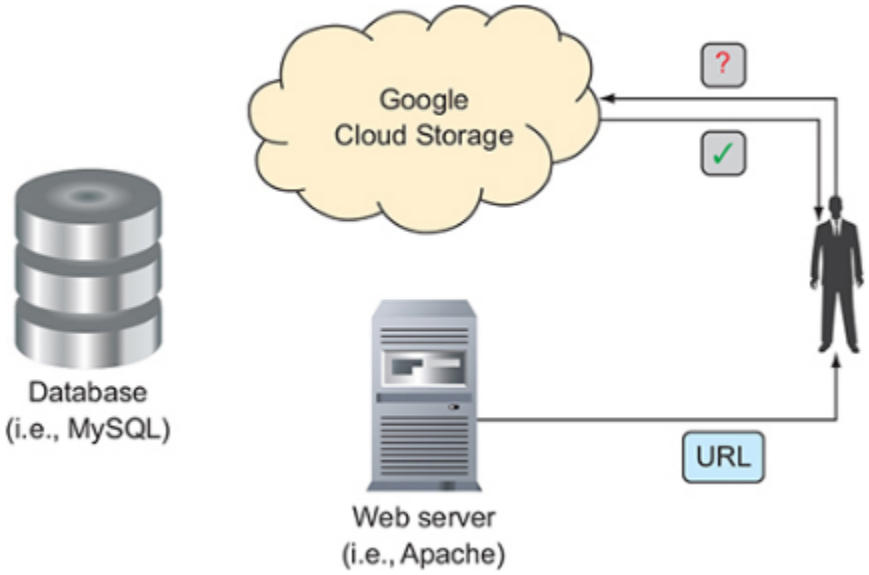

Figure 2. 2

2.2 Signing UP for GCP

In order to use GCP or Google cloud platform, we need to have a Google account such as Gmail account, with any account of Gmail, we would be able to login to Google Cloud at the following address; cloud.google.com. (Geewax, 2018)

When we have signed up, a new project is created automatically where the resources for one project would be isolated from other resources of other projects. (Geewax, 2018)

2.3 Project:

Projects in Google are containers of resources, API, billing, authentication and monitoring settings for those APIs related to that project. Also, it involves billing that any charges for resources are charged to the project. Projects would create some isolations in all aspects of a project, for instance, in-charge person of a project could not have an access to other projects unless specifically granted access. (61) (Geewax, 2018)

2.4 SDK

SDK is a suite of tools for building software by using Google cloud. It also involves tools for managing our resources in our projects. Generally, anything that we want to do with using Cloud console, could be done through SDK as well. In order to install SDK, first of all we need to go to this URL https://cloud.google.com/sdk/ and then follow the below instruction; (Geewax, 2018)

```
$ export CLOUD_SDK_REPO="cloud-sdk-$(lsb_release -c -s)"
$ echo "deb http://packages.cloud.google.com/apt $CLOUD_SDK_REPO main" | \
  sudo tee -a /etc/apt/sources.list.d/google-cloud-sdk.list
$ curl https://packages.cloud.google.com/apt/doc/apt-key.gpg | sudo \
  apt-key add -
$ sudo apt-get update && sudo apt-get install google-cloud-sdk
```

Figure 2. 3

Sometimes, we may get a message regarding to upgrading some components same as below capture; (Geewax, 2018)

```
Updates are available for some Cloud SDK components. To install
them, please run:
 $ gcloud components update
```

Figure 2. 4

In order to upgrade, we only need to run this command; “gcloud components update”.
After installing everything, we need to authenticate ourselves for Google. Google made it easy by connecting our terminal and browser;

```
$ gcloud auth login
Your browser has been opened to visit:

    [A long link is here]

    Created new window in existing browser session.
```

Figure 2. 5

A normal Google login and authorization screen would appear that asking us to grant the Google Cloud SDK access to our cloud resources. After clicking Allow, the window would close automatically and the prompt should update to look like the below capture; (Geewax, 2018)

```
$ gcloud auth login
Your browser has been opened to visit:

    [A long link is here]


Created new window in existing browser session.
WARNING: `gcloud auth login` no longer writes application default credentials.
If you need to use ADC, see:
  gcloud auth application-default --help

You are now logged in as [your-email-here@gmail.com].
Your current project is [your-project-id-here].  You can change this setting
     by running:
  $ gcloud config set project PROJECT_ID
```

Figure 2. 6

The below message maybe appeared that is related to credential of future application and code that would be the same as this credential. (Geewax, 2018)

```
$ gcloud auth application-default login
Your browser has been opened to visit:

    [Another long link is here]

Created new window in existing browser session.

Credentials saved to file:
     [/home/jjg/.config/gcloud/application_default_credentials.json]

These credentials will be used by any library that requests
Application Default Credentials.
```

Figure 2. 7

2.5 Interact with GCP

There are several ways that we can interact with GCP. First, we will create a virtual machine through a console, then from different ways, we would be able to connect to it.

In order to create a VM in console, we need to start by navigation to the Google compute Engine area of the console, then click on the Compute section in order to expand it, then click on the link

of Compute Engine. Once it is completed, a Create button would appear that by clicking on it, we would be able to adjust features of VM. (Geewax, 2018)

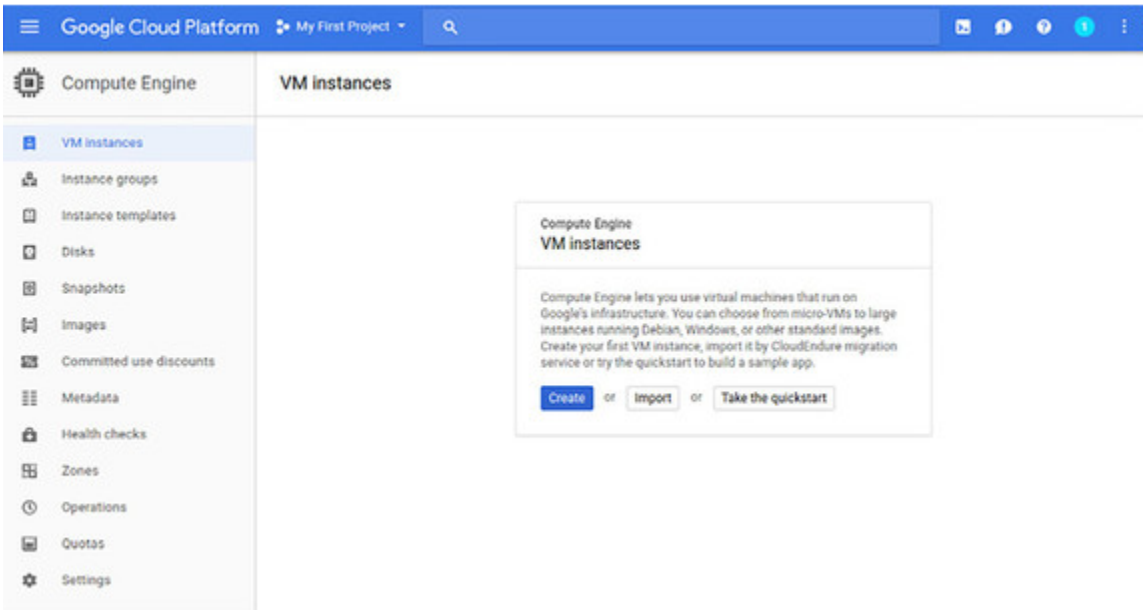

Figure 2. 8

On the Next Page, we would be able to configure all the details of our instance. We need to identify a name for our instance that should be unique in the whole project. Also, under zone field, we can choose where we want our instance to live. Next, in machine type, we could identify how our instance should be powerful. Google has several options of VM, ranging from f1-micro (which is a small, not powerful machine) all the way up to n1-highcpu-32 (which is a 32-core machine), or a n1-highmem-32 (which is a 32-core machine with 208 GB of RAM). After creating an instance, we could see a green checkmark in the list of instances in the console. (Geewax, 2018)

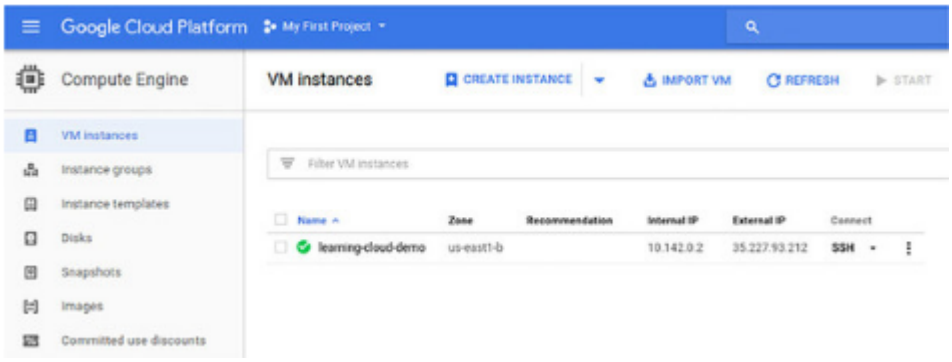

Figure 2. 9

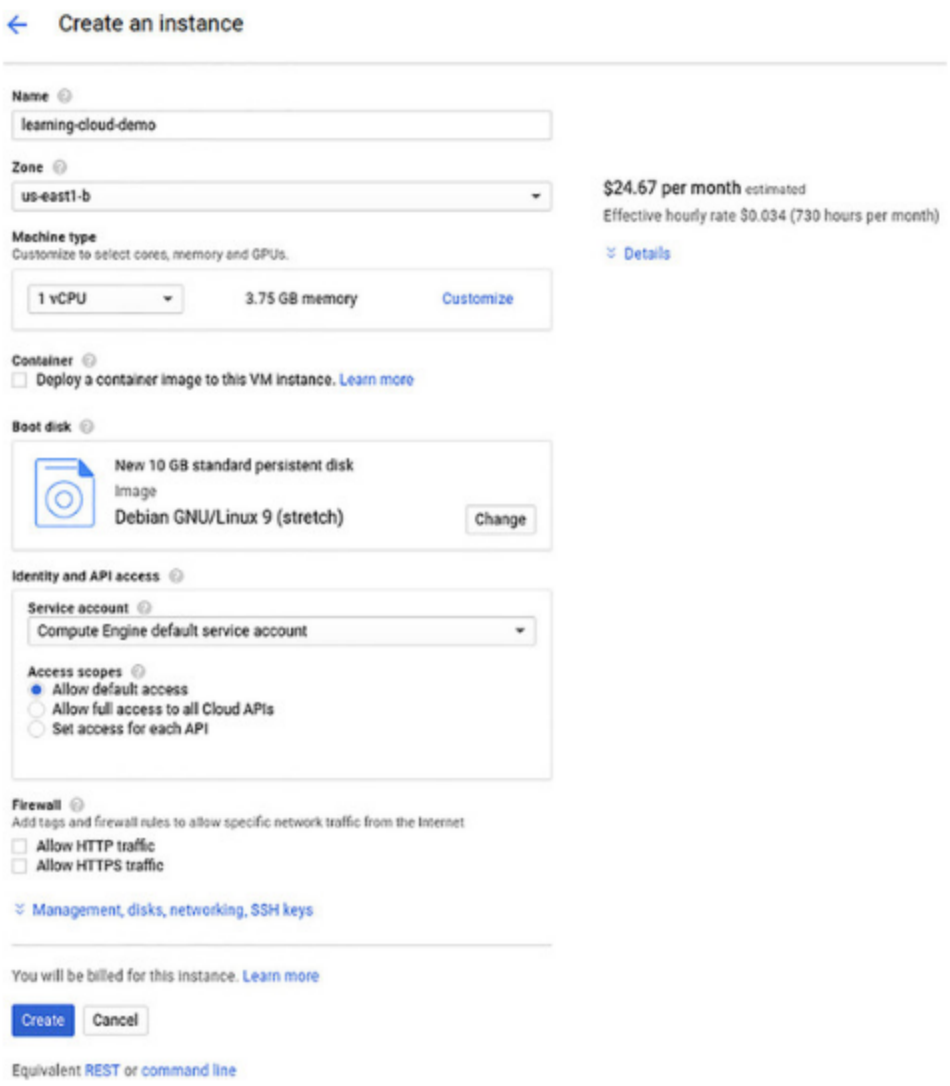

Figure 2. 10

As it is mentioned earlier, we can interact with all resources through SDK Tools, for instance, through below commands, we could be able to see a list of resources; (Geewax, 2018)

```
$ gcloud compute instances list
NAME                ZONE          MACHINE_TYPE  PREEMPTIBLE INTERNAL_IP
     EXTERNAL_IP   STATUS
learning-cloud-demo us-central1-b n1-standard-1            10.240.0.2
     104.154.94.41 RUNNING
```

Figure 2. 11

By typing “gcloud compute ssh “name of instance” and choosing the Zone, we would be able to connect to the instance via SSH. (Geewax, 2018)

We just need to consider that SDK is using the credential, which is prompted when we run "gcloud auth login" that generating new public and private keys. It puts the public key onto the virtual machine and using the private key in order to connect to VM automatically. It means that there is no need to worry about key pairs. (Geewax, 2018)

2.6 WordPress in Google Cloud

First of all, we need to know which process should be taken in order to create a webpage through a WordPress. (Geewax, 2018)

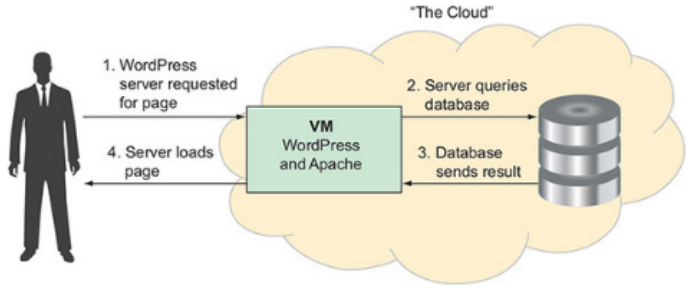

Figure 2. 12

As the diagram illustrates, first of all, end-user requests the WordPress server for a page then the server send this request to the database, the database checks this request and sends back the result to server, server sends this result to end-user.

As it is mentioned earlier, if we hosted our server in the US and a person from Asia sends a request to our servers, the request should traverse a long distance, so for this kind of case, we need to rely on the cloud solutions. The above diagram would change as below. (Geewax, 2018)

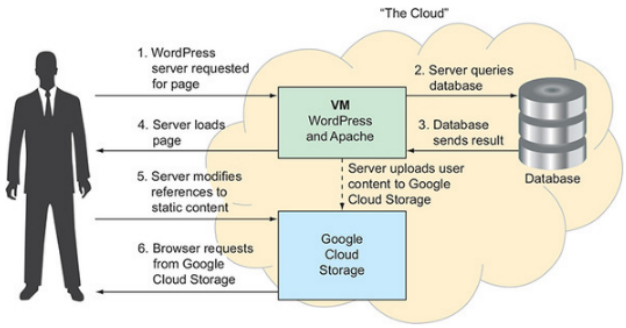

Figure 2. 13

For this case, the following would be the same at the beginning, but when static content is requested, our WordPress server modifies references to static contents, so the browser would request it from Google Cloud Storage. In this way, the geographic location of users is not

important. this means that requests for images and other contents would be handled by Google Cloud.

So, for this scenario, we need to select a database service. Since for most of the projects, a database service is required, so all common and famous cloud providers suggest some database services such as Relational Database service in Amazon, SQL database service in Azure and Google. Generally, managing a database via the cloud is somehow easier and quicker than configure and managing the underlying virtual machine and its software.

In order to turn on a database in Google, we need to create an instance of SQL through console. When a new instance is created, we need to configure some settings same as VM, such as name, password and location of SQL. Also, we can list all SQL databases through SDK tools with using “gcloud sql instance list” command the same as the below capture; (Geewax, 2018)

```
$ gcloud sql instances list
NAME          REGION  TIER              ADDRESS          STATUS
wordpress-db  -       db-n1-standard-1  104.197.207.227  RUNNABLE
```

Figure 2. 14

We could change password through command Line the same as the below capture;

```
$ gcloud sql users set-password root "%" --password "my-changed-long-
    password-2!" --instance wordpress-db
Updating Cloud SQL user...done.
```

Figure 2. 15

Moreover, we need to open a SQL instance to the outside. For this purpose, we can add network with 0.0.0.0/0 IP address range, the same as the below capture; (Geewax, 2018)

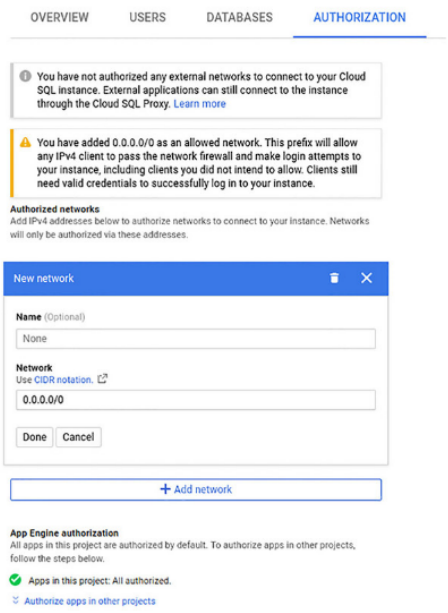

Figure 2. 16

The next step is installing the SQL Client in order to connect. On the Linux environment we could install it by typing the following code;

```
$ sudo apt-get install -y mysql-client
```

Figure 2. 17

For Windows and MAC, we need to download the package from MySQL website and use “root” as username, for password, we need to use the password which we defined in the previous step. At below, you would see the process on Linux;

```
$ mysql -h 104.197.207.227 -u root -p
Enter password: # <I typed my password here>
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 59
Server version: 5.7.14-google-log (Google)

Copyright (c) 2000, 2018, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input
     statement.

mysql>
```

Figure 2. 18

Right now, we need to prepare a SQL server for WordPress server, at below, you see the whole steps that should be taken; (Geewax, 2018)

- Create a database called “wordpress “ .

```
mysql> CREATE DATABASE wordpress;
Query OK, 1 row affected (0.10 sec)
```

Figure 2. 19

- Making a user with “wordpress” as name.

```
mysql> CREATE USER wordpress IDENTIFIED BY 'very-long-wordpress-password';
Query OK, 0 rows affected (0.21 sec)
```

Figure 2. 20

- Giving appropriate permissions for this user.

```
mysql> GRANT ALL PRIVILEGES ON wordpress.* TO wordpress;
Query OK, 0 rows affected (0.20 sec)
```

Figure 2. 21

Right now, we need to create an instance that could be a host for a WordPress installation. The steps would be the same as creating a normal instance, we just need to enable allowing traffic for HTTP and HTTPS ports because the WordPress server should be accessible to anyone through their browsers.

Figure 2. 22

At the next step, we need to connect to the VM via SSH, the same as the below capture;

```
$ gcloud compute ssh --zone us-central1-c wordpress
Warning: Permanently added 'compute.6766322253788016173' (ECDSA) to the list
     of known hosts.
Welcome to Ubuntu 16.04.3 LTS (GNU/Linux 4.13.0-1008-gcp x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  Get cloud support with Ubuntu Advantage Cloud Guest:
    http://www.ubuntu.com/business/services/cloud

0 packages can be updated.
0 updates are security updates.


The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

jjg@wordpress:~$
```

Figure 2. 23

After connecting, we need to install some packages such as MySQL client, PHP and Apache.

```
jj@wordpress:~$ sudo apt-get update
jj@wordpress:~$ sudo apt-get install apache2 mysql-client php7.0-mysql php7.0
     libapache2-mod-php7.0 php7.0-mcrypt php7.0-gd
```

Figure 2. 24

For next step, we need to install the WordPress;

```
jj@wordpress:~$ wget http://wordpress.org/latest.tar.gz
jj@wordpress:~$ tar xzvf latest.tar.gz
```

Figure 2. 25

Right now, we need to edit some parameters in WordPress. For example, we need to identify the location that WordPress should be stored the data and the way of authenticating. All these parameters should be done in “wp-config.php” file. (Geewax, 2018)

2.7 Zone

A zone is the smallest unit that a resource can exist. It is like a data center, but we need to consider an important point when we run two resources in the same Zone. If some natural disaster occurs in a zone such as a tornado, all the resources in that zone are likely going to

offline mode. The scenario would be the same for a malfunction inside a zone such as a power outage occurred. (Geewax, 2018)

2.8 Region:

A collection of zones is called a region and the distance between zones inside a region would be close to each other. If we run our resources into a single region but different zone, they would be isolated from zone-specific failures like a power outage, but they might not be isolated from catastrophes like a tornado. (Geewax, 2018)

2.9 Fault Tolerance

Google cloud offers a different level of isolation.

- Zonal: a zonal service means that if that zone goes down, the service goes down as well.
- Regional: a regional service means that a service is replicated through a different zone in a single region. So, the service would be isolated from malfunction issues. If one service in one zone goes down due to power outage, the same service in another zone would respond automatically.
- Multiregional: A multiregional service is a composition of several different regional services, it means that if one service in one region went down due to catastrophe reason, the same service in another region would respond to all our requests automatically. (Geewax, 2018)

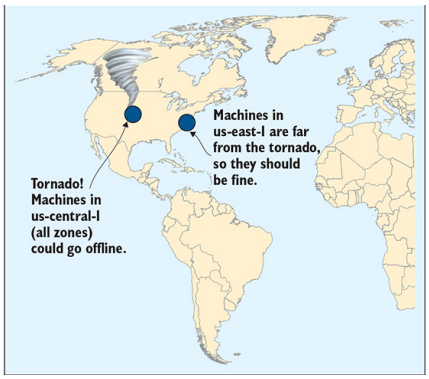

Figure 2. 26

- Global: A global service is a special case of multiregional service. With this option, we would be able to deploy a service in multiple regions, but these regions are spread around the world. At this point, we probably need to use multiple cloud providers. (Geewax, 2018)

2.10 Cloud SQL

Cloud SQL is a VM that is hosted in Google Cloud and running a version of the MySQL binary. First, we need to create an instance of SQL in the cloud by clicking on the SQL section in the cloud console and then click on the button to create a new instance.

We need to identify an instance ID and root password. Also, it is essential to select a region that this region should be close enough to our customers otherwise the queries have to around the world and back. (Geewax, 2018)

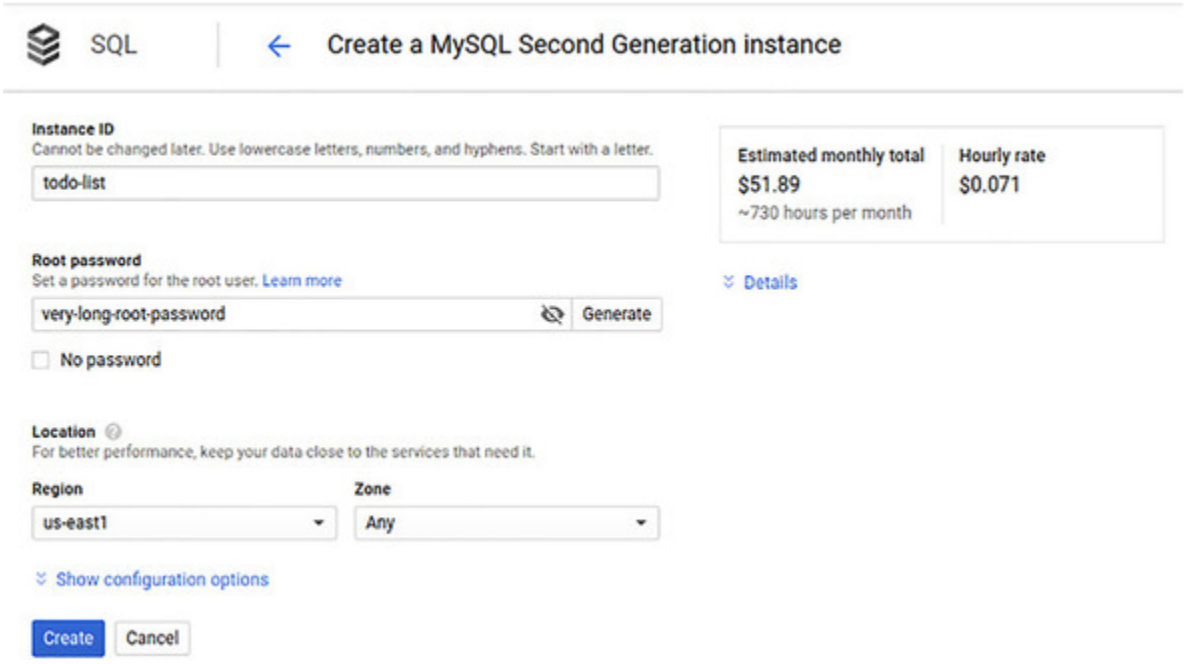

Figure 2. 27

In order to connect to SQL, we need a user. We would be able to create a new user in next tab "USERS", the same as the below capture; (Geewax, 2018)

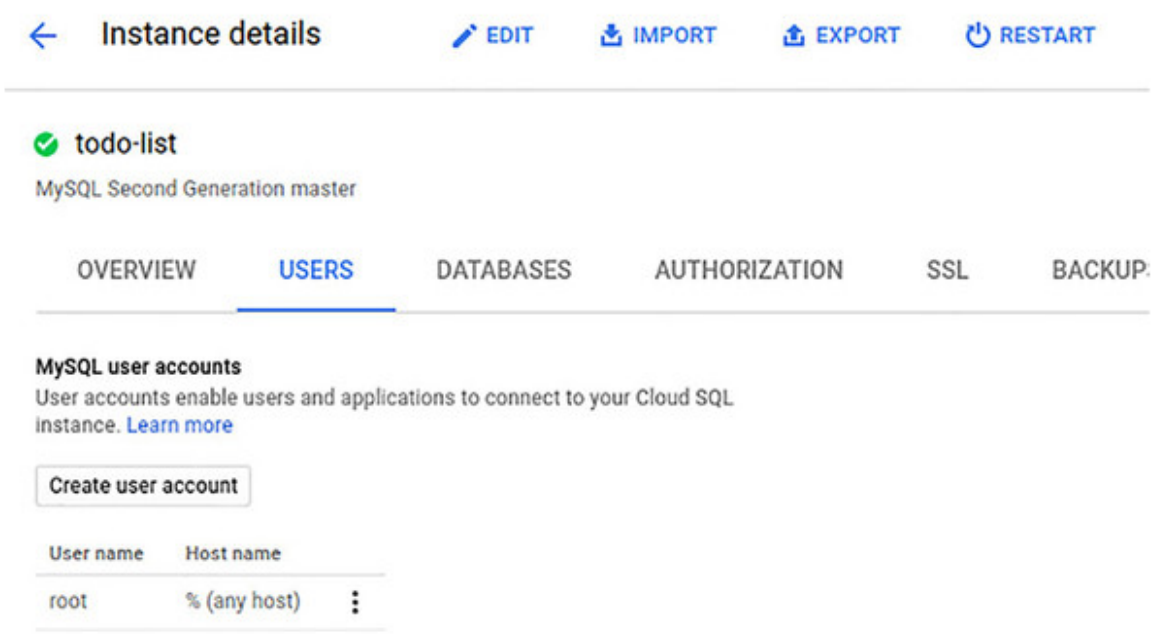

Figure 2. 28

After completing all the above steps, we need to go to the command line environment and start to talk to SQL instance; (Geewax, 2018)

```
$ mysql -h 104.196.23.32 -u user-here \
        --password=password-here                                        1
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 37
Server version: 5.6.25-google (Google)

Copyright (c) 2000, 2015, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

Figure 2. 29

Right now, we are ready to do whatever we need in SQL.

The next step is controlling access to our SQL. We can identify the range of IP address who can see our server by adding in the network section, the same as the below capture; (Geewax, 2018)

Figure 2. 30

Right now, we should create a secure connection between server and SQL, for this purpose we need to prepare three following items;

- The server's CA certificate: In order to create a CA server, we can go to SSL of SQL section, then click on the "View Server CA certificate" button then download the "server-ca.pem" file, the same as the below

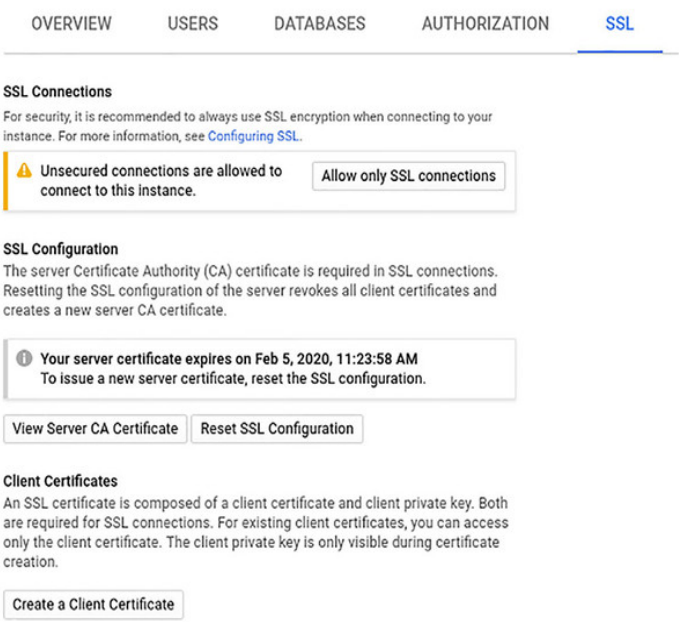

Figure 2. 31

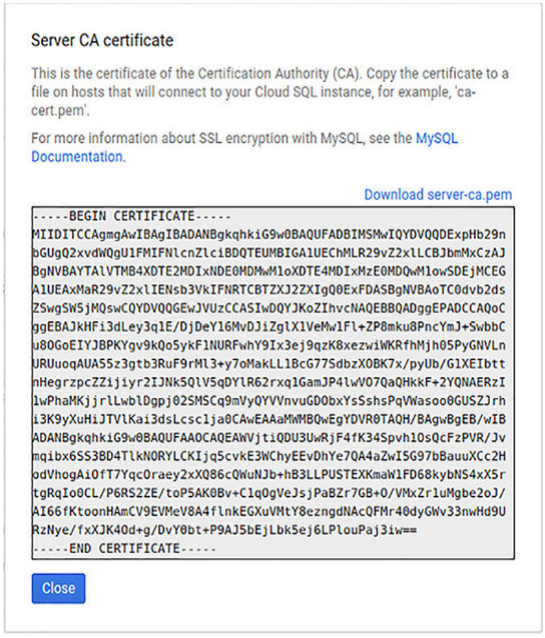

Figure 2. 32

- Client certificate: In order to create a client certificate, we can click on the “create a client certificate” button and input a name for that. (Geewax, 2018)

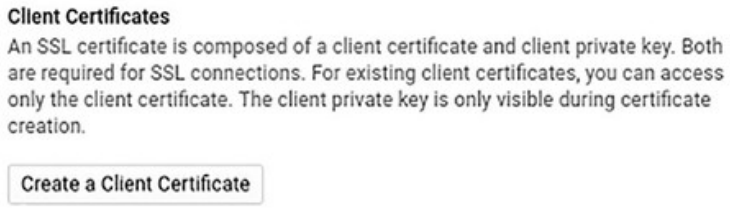

Figure 2. 33

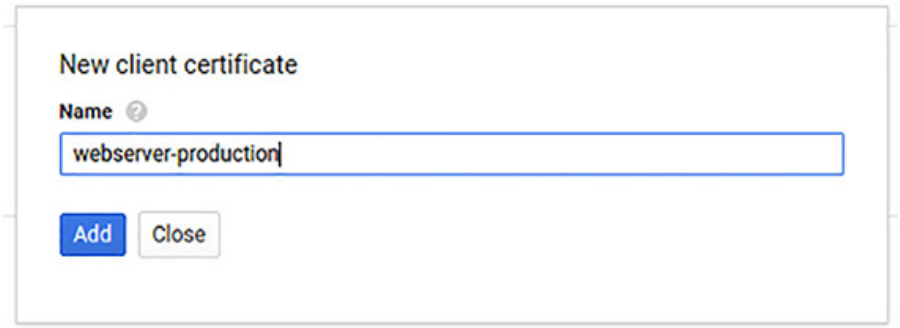

Figure 2. 34

- A client private key: When we click on the “add” in the previous section, the private key would be created automatically as well.

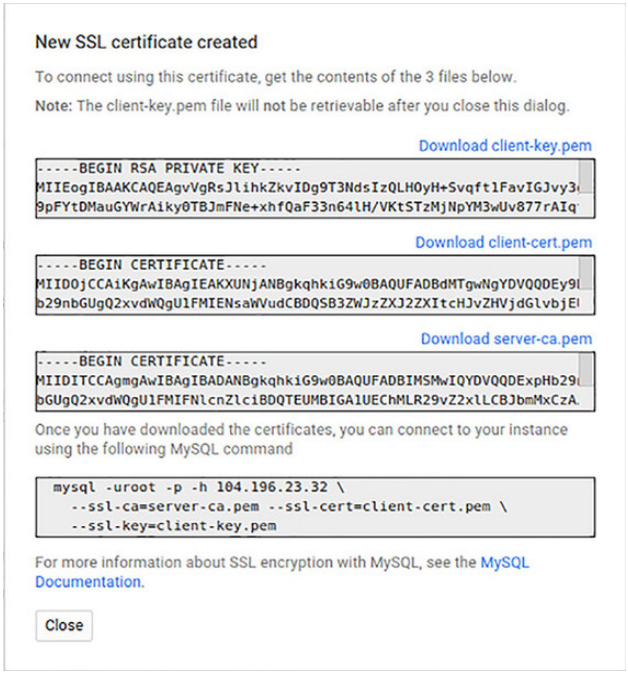

Figure 2. 35

There is a point that we need to consider. If we lose files “server-ca.pem” and “client-cert.pem”, we could be able to download then again but this case will not come true for the private key file. At the next step, we need to add these files in our SQL in order to make an SSL connection; By using the below command, we could make sure that our connection is secure;

```
$ mysql -u root --password=really-strong-root-password -h 104.196.23.32 \
  --ssl-ca=server-ca.pem \
  --ssl-cert=client-cert.pem \
  --ssl-key=client-key.pem

Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 646
Server version: 5.6.25-google (Google)

Copyright (c) 2000, 2015, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

Figure 2. 36

As a next step, we need to schedule sometimes for upgrading or updating security patches. All the maintenance would be done by Google Cloud, we just need to determine date and time. For this purpose, we need to go to the detail page of our instance and click on “Edit maintenance schedule” link at the bottom; (Geewax, 2018)

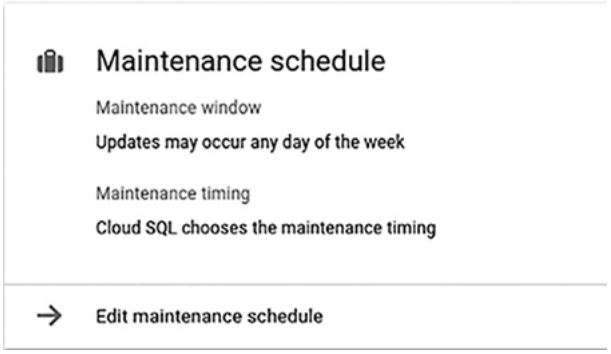

Figure 2. 37

Then we need to set a schedule for maintenance; we need to identify a day of week and time. Just regarding maintenance timing, if we set on later, it means that on the next period the latest version would be applied but the earlier timing means that our instance would be upgraded as soon as the newest version is considered stable. (Geewax, 2018)

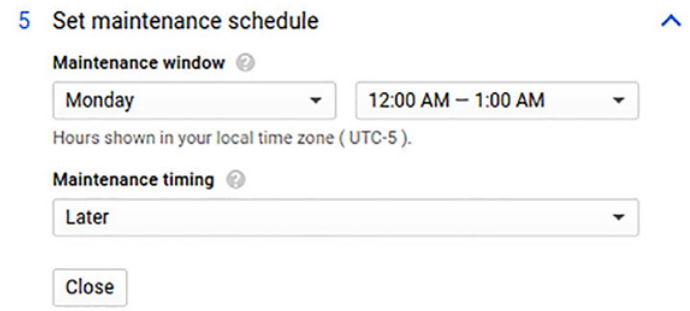

Figure 2. 38

It is also possible to tune MySQL for maximum performance by using Add Database Flags option from the configuration section under each SQL instance, the same as the below capture;

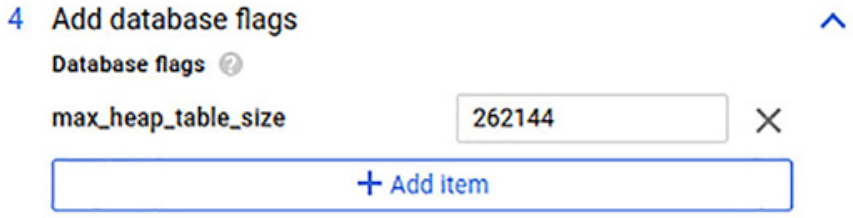

Figure 2. 39

Another point that we need to consider is that SQL performance is depended on computing power and disk. We can change type of instance by clicking on “Edit” button in the detail page of Instance; (Geewax, 2018)

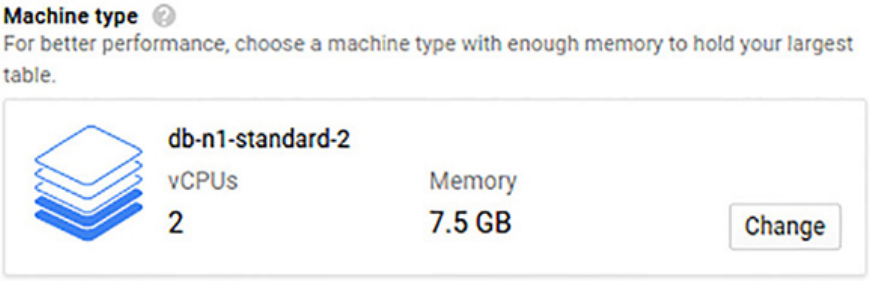

Figure 2. 40

But after changing machine type, once reboot is needed. Also, through the same menu, we would be able to increase the capacity of disk; (Geewax, 2018)

Figure 2. 41

Another aspect for SQL is replication. Google Cloud has two kinds of replication scenario. First is read replica DB that actually is a clone of our Cloud SQL instance. If any failure occurs for master DB, it would take over all responsibilities of master except INSERT and UPDATE quarries, because it can do read-only. As a result, read replicas would be useful when our application needs a lot of more read queries than writes. This kind of replication would increase the scale of our system horizontally. (Geewax, 2018)

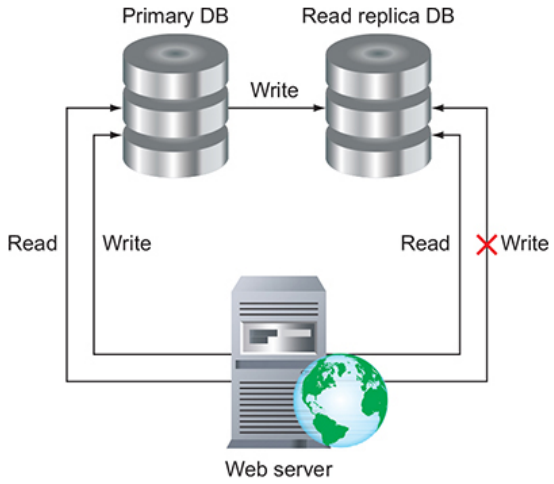

Figure 2. 42

Second is failover replica that is the same as a read replica but it would be exactly a replacement of primary instance in case of any kinds of failure for the master DB. (Geewax, 2018)

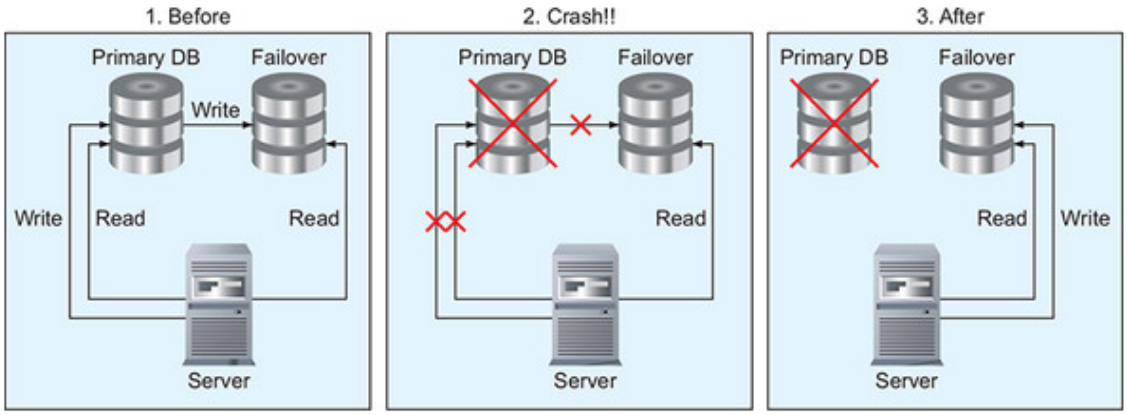

Figure 2. 43

In order to create these replicas, we need to go to the console of Google cloud then navigate over to the list of SQL instances and click on the button “Add Failover”. By choosing Failover

replica, we would be able to choose a different zone, but in the same region and by clicking on the read replica, we would be able to choose a different machine type. (Geewax, 2018)

2.11 Document Storage

In this kind of database, all data would be stored in the form of collections and documents rather than keeping all data in a rectangular grid. These kinds of documents are arbitrary sets of key-value pairs and the document types must be matched to each other. Given below, the same data has been shown. The first capture is related to a traditional table and the second one shows the same data in a document collection's jagged format; (Geewax, 2018)

ID	Name	Favorite color
1	"James Bond"	Null
2	"Ian Fleming"	"blue"

Figure 2. 44

Key	Data
1	{id: 1, name: "James Bond"}
2	{id: 2, name: "Ian Fleming", favoriteColor: "blue"}

Figure 2. 45

2.12 Cloud Datastore

Cloud Datastore is called App Engine Datastore that came from a storage system Google built called Megastore.it is designed to handle large scale data. One the purpose usage of Datastore is that we can choose which documents live near to other documents by putting them in the same entity group. Another matter is that, if we run a query in a database the time to respond is dependent to the scale of data while datastore by using indexing accomplish this case. It means that if our query has 10 matches, it will take the same amount of time regardless of whether we have, 1 GB or 1PM data. Also, datastore can take care when data is distributed over several places and it needs to retrieve data from several servers over the world.

there are some differences between a datastore and a relational database:

- Key: Data store is using keys that contain both types of data and data's unique identifier. Also, keys could be hierarchical, which is a feature of the concept of data locality. If two keys have the same parent, it means that they are in the same entity group. Actually, with using parent key, we would be able to find the location that data should be put.

- Storage: the primary storage in datastore is entity. An entity is nothing more than a collection of properties and values that combined with a unique identifier called a key.
- Operation: the basic operations in datastore are;
    - Get- retrieve an entity
    - Put- save or update an entity
    - Delete- delete an entity

  The key of the entity is needed for all these operations.
- Indexing and Queries: In a typical database, a query is nothing more than a SQL statement while in Datastore that would be possible through GQL. Datastore looks like a SQL but there are some queries that Datastore cannot answer. Actually, relational database use indexing as a way of optimizing a query whereas Datastore uses indexes to make a query possible. (Geewax, 2018)

Feature	Relational	Datastore
Query	SQL, with joins	GQL, no joins; certain queries impossible
Index	Makes queries faster	Makes advanced query possible

Figure 2. 46

2.12.1 Consistency and replication

A distributed storage system needs to meet two keys, availability and scale with result set. This means that not only data should be replicated but also needs to create and maintain indexes for the queries. For data replication, cloud Datastore uses one protocol that called a two-phase commit. In this protocol we would have two phase for data replication, a preparation phase that we sent a request to a set of replicas and describe the changes that we want and ask them to get ready to apply, while all replicas confirm that they are ready to apply we send a second request instructing all replicas to apply that change. This method leads to eventual consistency when running broad queries where the entity or the index entry may be out of date. Any consistent query will push replicas to execute any pending commits then run the query. So, maintaining entities and indexes in a distributed system is a complicate task because any save operation would result to save to any indexes that the changes affect. It means that datastore would have two options; (Geewax, 2018)

- Update the entity and the indexes everywhere synchronously.
- Update the entity itself then update the indexes in the background.

As we mention earlier, Datastore chose to update data synchronously and the time needs to save several indexes would be the same as one. In result, when we use put operation, under the hood Datastore will do quite a bit of work;

- Create or update the entity
- Identify which indexes need to change
- Send a request to all replicas in order to prepare for changes
- Ask the replicas to apply the changes

After a consistence query runs;

- Ensure all pending changes to the affected entity group are applied
- Execute the query (Geewax, 2018)

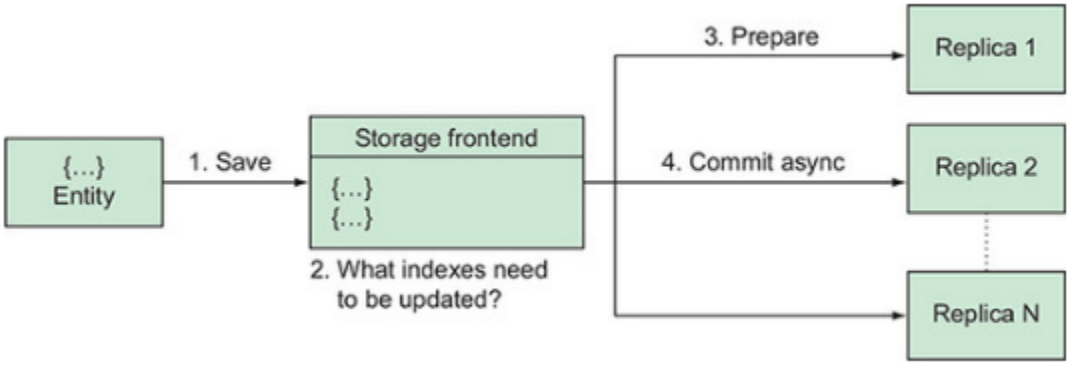

Figure 2. 47

It means when we run a query, Datastore uses these indexes to run a query in time that is proportional to the number of matching results found. So, query will do the following;

- Send a query to Datastore
- Search indexes for matching keys
- For each matching result, get the entity by using its key
- Return the matching entities

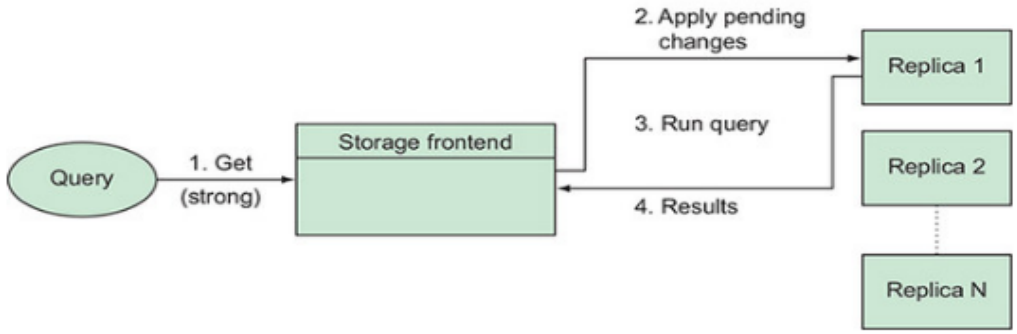

Figure 2. 48

The key piece is that the indexes would be updated in the background but there is no guarantee regarding when the indexes would be updated. This concept is called eventual consistency. Actually, eventually the indexes would be up to date which data that you have stored in the entity. But it has an issue which explained through the below example;
Consider that we want to add a new employee entity to cloud Datastore, as shown in the following capture; (Geewax, 2018)

```
{
  "__key__": "Employee:1",
  "name": "James Bond",
  "favoriteColor": "blue"
}
```

Figure 2. 49

Now when we want to select all employees with blue as their favorite color;

```
SELECT * FROM Employee WHERE favoriteColor = "blue"
```

Figure 2. 50

If their indexes have not been updated yet because of eventually update, we will not see that entity in our result while if we ask specifically for the entity, it would be there.

```
get(Key(Employee, 1))
```

Figure 2. 51

The same scenario would apply for modification as well. For example, imagine that the indexes have reached a level of consistency when we are looking for employees with blue as their

favorite color and before that we changed the favorite color of one of entity the same as the below;

```
{
  "__key__": "Employee:1",
  "name": "James Bond",
  "favoriteColor": "red"
}
```

Figure 2. 52

So, if we run our query again, based on which update has been committed we would see different result as described in the below capture; (Geewax, 2018)

Entity updates	Index updated	Employee matches	Favorite color
Yes	Yes	No	Doesn't matter
No	Yes	No	Doesn't matter
Yes	No	Yes	red
No	No	Yes	blue

Figure 2. 53

In order to figure out how this design would benefit customer like Gmail, we need to explain the concept of combining queries with data locality to get strong consistency. (Geewax, 2018)

2.12.2 Consistency with data Locality

Data locality is a tool for putting many pieces of data near to each other. First of all, we need to explain that queries inside a single entity group are strongly consistent not eventually consistent. Actually, it means that if we run a query over a bunch of entities that all have the same parent keys, our query would be strongly consistent.

Running query over in terms of locality provides us a feature that make sure all pending operation on those entities would be completed first then the query would run. In this case, all the time we would have the updated results. (Geewax, 2018)

2.12.3 Interacting with cloud Datastore

Before using Cloud Datastore, we need to enable this feature in Google Cloud by searching "Cloud Datastore API" through the cloud console main search box and then clicking on the "Enable" button the same as the below capture. If there is only "Disable" button, it means that we already enable it. (Geewax, 2018)

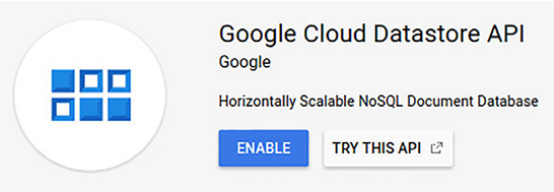

Figure 2. 54

It should be mentioned that unlike SQL, first of all, we need to create some data rather than defining a schema. Actually, this is the nature of nonrelational storage. In order to add data, first, we need to create an entity that would be possible by clicking on the "Create Entity" button when we visit the Datastore page. As it could be seen in the below figure, we created an entity with name "TodoList" and added a property with String as type and Grocery as value. Since we want to run a search based on this name, we need to leave the property indexed by marking the checked box. (Geewax, 2018)

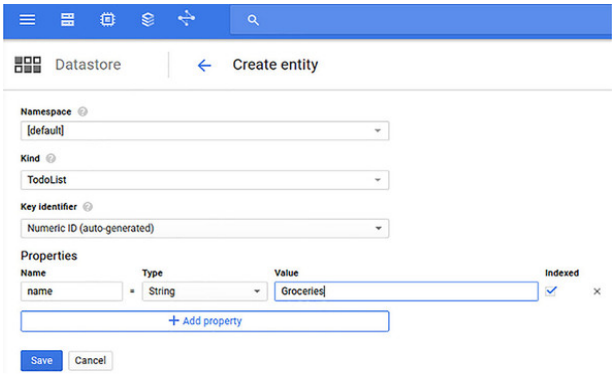

Figure 2. 55

When we click on the save, we would see the new entity in our bowser like the below one.

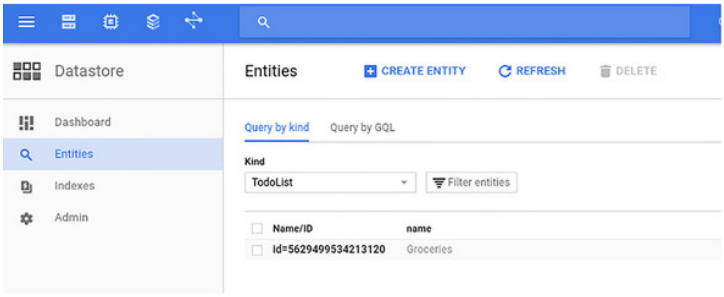

Figure 2. 56

2.12.4 Backup and Restore

Backup in Datastore is different from that one we get used to it. Actually, Datastore's eventually consistent queries make it difficult to get the overall state of data at a single point in time. Because of this effect, we need to remember that exports would not be a snapshot that taken at a

single point in time. Indeed, they would be like a long-exposure photograph of our data. In order to minimize the effects of this long exposure, we need to disable Datastore writes beforehand and then re-enabling them once the exports complete. In order to complete backup operation, we need to create a cloud Storage bucket which is a place that holds our exported data, the same as the below capture. (Geewax, 2018)

```
$ gsutil mb -l US gs://my-data-export
Creating gs://my-data-export/...
```

Figure 2. 57

After that we need to disable writes to our Datastore instance through Cloud Console by using Admin tab in Datastore section. (Geewax, 2018)

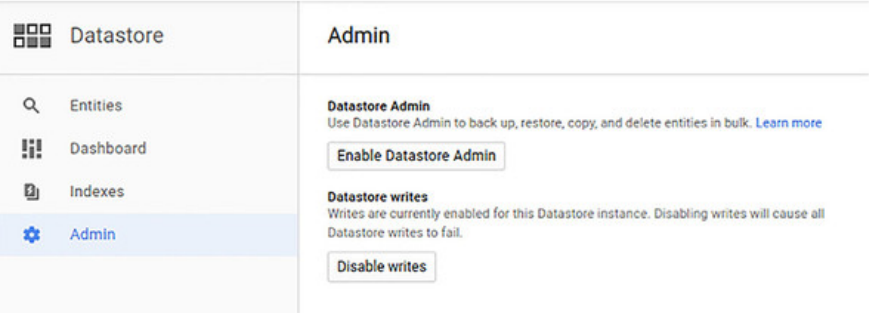

Figure 2. 58

As the last step, we need to trigger an export of data into our bucket through Datastore export subcommand the same as the below capture. (Geewax, 2018)

```
$ gcloud beta datastore export gs://my-data-export/export-1
Waiting for [projects/your-project-id-here/operations/
     ASA1MTIwNzE4OTIJGnRsdWFmZWQHEmxhcnRuZWNzdS1zYm9qLW5pbWRhFAosEg] to
     finish...done.
metadata:
  '@type':
     type.googleapis.com/google.datastore.admin.v1beta1.ExportEntitiesMetadata
  common:
    operationType: EXPORT_ENTITIES
    startTime: '2018-01-16T14:26:02.626380Z'
    state: PROCESSING
  outputUrlPrefix: gs://my-data-export/export-1
name: projects/your-project-id-here/operations/
     ASA1MTIwNzE4OTIJGnRsdWFmZWQHEmxhcnRuZWNzdS1zYm9qLW5pbWRhFAosEg
```

Figure 2. 59

Then we need to verify data arrived in the bucket by using the below command;

```
$ gsutil du -sh gs://my-data-export/export-1
32.2 KiB    gs://my-data-export/export-1          1
```

Figure 2. 60

Regarding restoring, we need to consider several points. First of all, importing entities would use all the same IDs as before. Indeed, it overwrites any entities that use those IDs. Secondly, all entities that we created after export entities, will remain because import could edit and create entities but will not never delete any entities. In order to run an import, the same as exporting, first, we need to disable rewrite then doing import by pointing to metadata file that was created during export, the same as the below capture. (Geewax, 2018)

```
$ gsutil ls gs://my-data-export/export-1                          1
gs://my-data-export/export-1/export-1.overall_export_metadata     2
gs://my-data-export/export-1/all_namespaces/
```

Figure 2. 61

Then doing the import job by using the below command.

```
$ gcloud beta datastore import gs://my-data-export/export-1/export-
    1.overall_export_metadata
Waiting for [projects/your-project-id-here/operations/
    AiA4NjUwODEzOTIJGnRsdWFmZWQHEmxhcnRuZWNzdS1zYm9qLW5pbWRhFAosEg] to
    finish...done.
metadata:
  '@type': type.googleapis.com/google.datastore.admin.v1beta1.ImportEntitiesMetadata
  common:
    operationType: IMPORT_ENTITIES
    startTime: '2018-01-16T16:26:17.964040Z'
    state: PROCESSING
  inputUrl: gs://my-data-export/export-1/export-1.overall_export_metadata
name: projects/your-project-id-here/operations/
AiA4NjUwODEzOTIJGnRsdWFmZWQHEmxhcnRuZWNzdS1zYm9qLW5pbWRhFAosEg
```

Figure 2. 62

2.13 Large-scale SQL

SQL database is very rich in queries, consistency and transactional semantics but has a problem in handling a massive amount of traffic. NoSQL (none-relational SQL database) has some or all these strong points in exchange for horizontal scalability which allow systems to handle more traffic by adding more machines to a cluster easily. So, there was a need to introduce a new SQL which has rich queries, transactional semantics, strong consistency and horizontal scalability which is called NewSQL database. Actually, it acts like SQL database, but it has a scaling feature of NoSQL database. In more explanation, New SQL could have data locality but still we can make query using familiar SELECT * FROM syntax. (Geewax, 2018)

2.14 Spanner

Spanner is combining the scalability feature of nonrelational storage with a traditional MySQL database. Actually, spanner is a NewSQL database which provides most of features of a relational database such as schemas and JOIN queries with a scaling option of nonrelational database like being able to add more machines. In case of large load, spanner would split data across more machines. Spanner has strongly consistent queries so it will not have a stale version of the data. In the following, we will go through the concepts of cloud spanner. (Geewax, 2018)

2.14.1 Instance

A cloud spanner instance acts as an infrastructural container which holds a bunch of databases. It manages several discrete units of computing power which is responsible to serve spanner data. Spanner instances have two aspects, data-oriented aspect and infrastructural aspect. First, we will explore in data-oriented aspect. When we run a query and receive the results, an instance would act like a database container. But regarding infrastructural aside, spanner instances automatically replicate unlike MySQL instances. Indeed, in this case, instead of choosing a specific zone where instance will live, we choose some configuration that combine different zones. For instance, the regional-us-central configuration means the combination of zones which live inside us-central region. (Geewax, 2018)

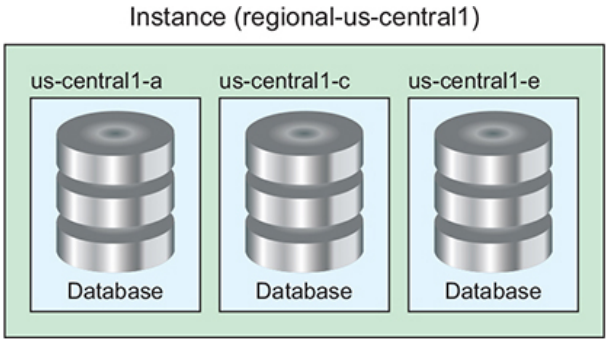

Figure 2. 63

2.14.2 Node

Spanner instances are made up of specific nodes that used to serve instance data. Those nodes live in specific zones and responsible to handle queries. At the same time, spanner instances are replicated completely so if any zone goes to down state because of any reason, we will make sure that data will continue serving without any problems. (Geewax, 2018)

For instance, if we have three nodes in each regional, we will have a total of nine nodes because each replica is a copy of both data and the serving capacity. With this method, spanner guarantees rich query, strong consistency, high availability and performance. (Geewax, 2018)

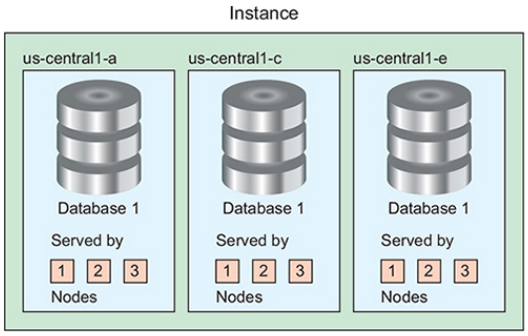

Figure 2. 64

2.14.3 Databases

Databases are tables. A single database is a container of data for a single product. Databases are used to make schema changes and query for data. From most of the aspects, spanner tables are similar to other relational databases but with some differences. First, we will explore what would be the same and what would be the differences between spanner tables and relational databases. Tables would have a schema; they would have columns with type for each column and modifiers. Like normal relational databases, adding data that does not meet type defined in schema results in error. In the below capture, you would see creating a table in the relational database; (Geewax, 2018)

```
CREATE TABLE employees (
  id INT NOT NULL AUTO_INCREMENT PRIMARY_KEY,
  name VARCHAR(100) NOT NULL,
  start_date DATE
);
```

Figure 2. 65

While in order to create the same table in spanner we need to use the below command;

```
CREATE TABLE employees (
  employee_id INT64 NOT NULL,
  name STRING(100) NOT NULL,
  start_date DATE
) PRIMARY KEY (employee_id);
```

Figure 2. 66

So, the only differences are the location of primary key and data type name.

2.14.4 Creating spanner

In order to create a spanner, first we need to enable cloud spanner API by clicking on the "Enable" Button. (Geewax, 2018)

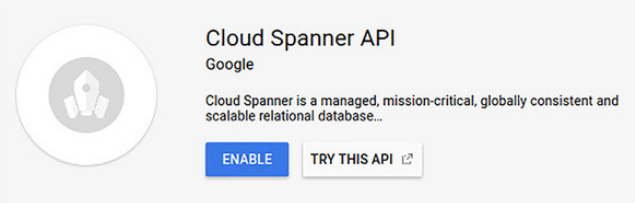

Figure 2. 67

The next step is creating spanner instance same as the below capture; (Geewax, 2018)

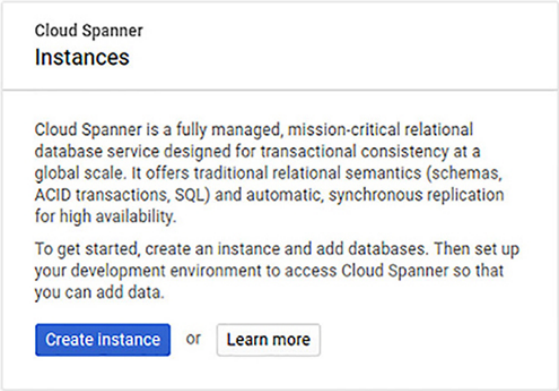

Figure 2. 68

By clicking on the "Create instance" button, we need to input some more details such as name, location and so on. (Geewax, 2018)

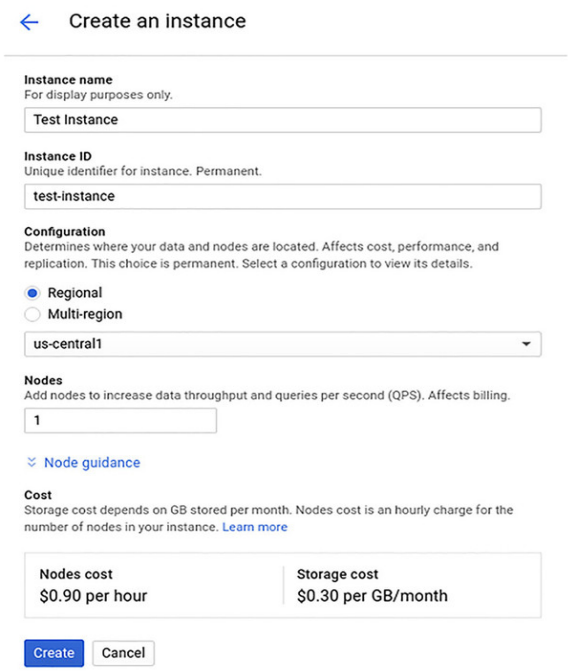

Figure 2. 69

In the next step, we need to create a new database for this instance,

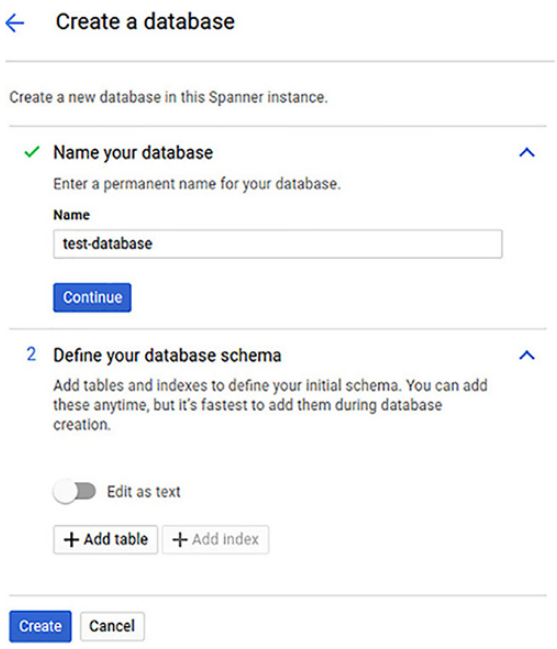

Figure 2. 70

So, we should be able to see the same as the below capture in our page;

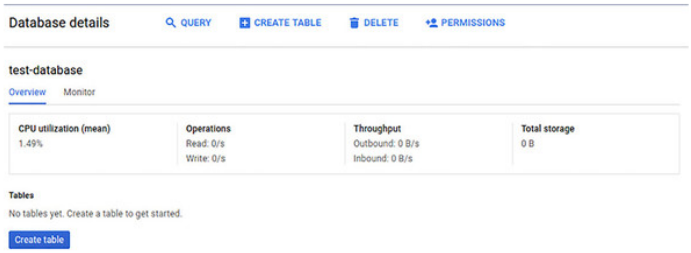

Figure 2. 71

The next step is creating a table. As it is mentioned earlier, most table features would be the same with the rational database. We would use text option in order to create a table,

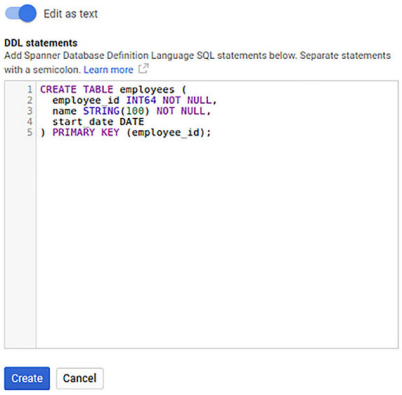

Figure 2. 72

After clicking on the “Create” button, we would see all the details such as schema, indexes, preview of data the same as the below capture;

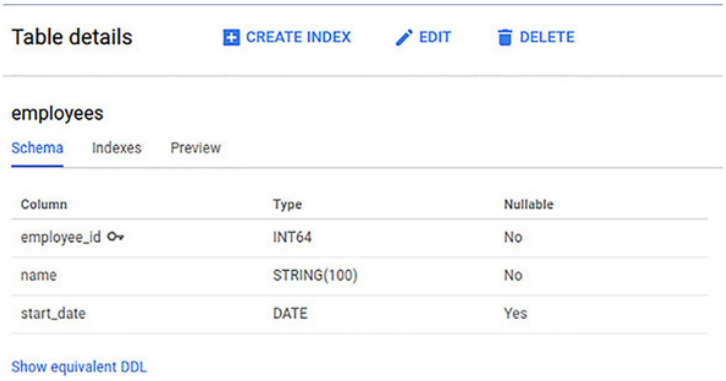

Figure 2. 73

One the main differences between the spanner and the rational database is the way that we could modify data. In the relational database, we would use INSERT query to add data and UPDATE in order to modify or update data while the spanner does not support these two commands. In the spanner, we could add data through a separate API which would be more similar to a non-

relational key-value system, where we choose a primary key and some values for that key. In order to run a query through the spanner, there are two ways, first by using spanner's Read API to query a single table, these kinds of queries would be looking up based on specific key or scanning through a table with some filters applied, or by running a SQL query which would be able us to query multiple tables using joins and other advanced filtering techniques. Regarding first way, it should be mentioned that query would be run over a database instead of only one table, also instead of sending a structured object to represent the query, we would send a string containing our SQL query. For instance, the below capture shows an example through reading API in order to fetch data from a table; (Geewax, 2018)

```
const spanner = require('@google-cloud/spanner')({
  projectId: 'your-project-id'
});
const instance = spanner.instance('test-instance');
const database = instance.database('test-database');
const employees = database.table('employees');
const query = {
  columns: ['employee_id', 'name', 'start_date'],
  keys: ['1']
};

employees.read(query).then((data) => {
  const rows = data[0];
  rows.forEach((row) => {
    console.log('Found row:');
    row.forEach((column) => {
      console.log(' - ' + column.name + ': ' + column.value);
    });
  });
});
```

Figure 2. 74

2.14.5 Advanced Concept

Regarding cloud spanner, we need to understand a little more how it would be combined together with a traditional relational database with a large-scale distributed storage system. So, we need to look at the schema level concept of interleaving tables with another one. (Geewax, 2018)

2.14.6 Interleave tables

Indeed, cloud spanner supports additional relational aspects which means combining two tables together, but we need to know what would be happened when we would have a heavy load. When we have a large number of requests for data, a single server cannot handle it. The first step is adding servers that duplicate data and act as alternative servers to query data. This solution would be the best one when we have a large number of requests for reading data rather than writing data. Because regarding writing data, we need to write data in all servers, the same as the below figure, (Geewax, 2018)

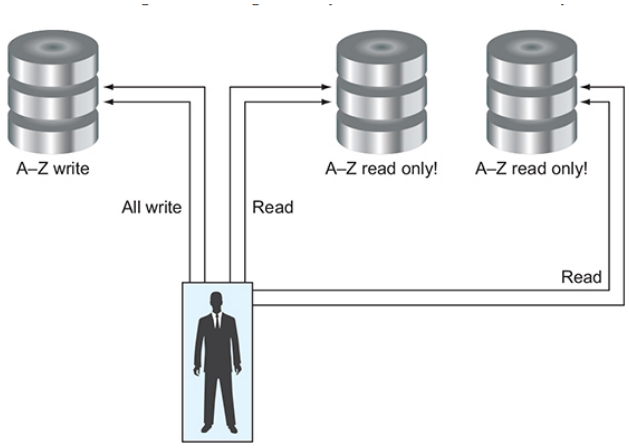

Figure 2. 75

In this kind of situation, one solution would be share data across multiple machines. Actually, we could chop up data into distinct pieces and delegate responsibility for different chunks to different machines. For instances, regarding an employee table, we could save employees who their names start from A through L on one server and remains names in another server. So, each query would handle by one server and actually we are spreading loading on two servers. Frankly speaking, we will use interleaving tables in the spanner in order to identify where data should live. Regarding primary key, in the spanner, we could have a primary key which consists of a combination of several fields. (Geewax, 2018)

2.15 Bigtable

Bigtable has been started as the storage system for the web search index at Google. Actually, the main purpose of this storage system is to solve a specific but complex problem in which it could update petabyte of data with incredibly high throughput, low latency and high availability. This problem cannot be solved by using MySQL because it falls over quickly in attacking. So, Google needs another way by using a globally key-value map, which automatically rebalances data in order to reach performance. The main purpose of Bigtable is for web search indexes. Two requirements are needed for this goal, performance and scale. First, we need to understand these two requirements. It is needed to mention that search indexes would be enormous with overall size measured in petabytes. It means that is too large for a server to manage. So, there would not be an option except we split data over different servers. But at the same time, we have to make sure that all piece of data would be available all the time. Also, the search index clearly sees a ton of traffic, potentially millions of queries every second. Each search query should reply a respond quickly within in millisecond. Another item that should be considered is that data is

rapidly changing. New web pages are added all the time and the search index would be updated by a web frequently. Regardless of the number of queries, the system should be able to handle so many updates at the same time. We need to keep the history of changing data. For this case, we need a way to see the data as it was at a particular time. The client would done this manually by constructing keys with timestamps to identify which version of data we are referring. In order to keep track of changes, we would create three-dimensional system the same as the below figure. With this ability, we would be able to find the latest value in a row, as well as all the values that this row has had over time. (Geewax, 2018)

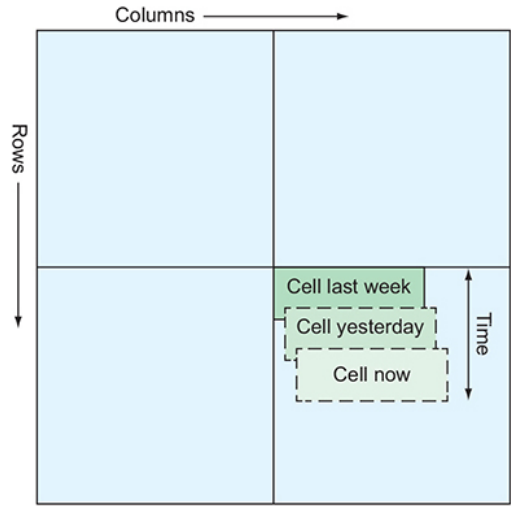

Figure 2. 76

The next one is strong consistency which means whenever query is received, the respond would be the latest version of data. If the system does not have this feature, we may have different results of searching same data in two browsers at the same time.
Moreover, the system needs to allow atomic read-modify-write sequence or risk two updates overwriting each other if we want to have a system that shows a consistent view of the world. The system should return an error if someone else has changed a row data while we are attempting to work on it. The below figure shows what would be happened when two write data overlap each other. (Geewax, 2018)

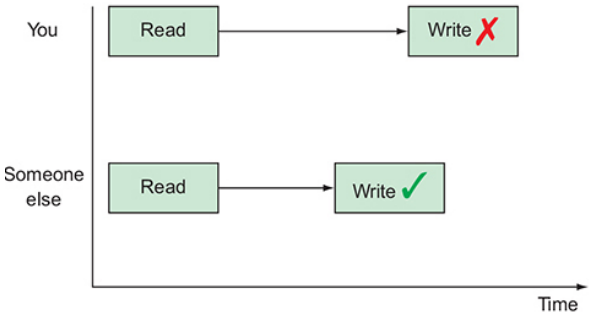

Figure 2. 77

3. Microsoft Azure

3.1 Resources

Resources are the smallest entity in Azure. Virtual machines, storage accounts, web apps and so on are examples of resources in Azure. Each resource should be deployed in one resource group while a resource group could be included several resources. It should be mentioned that each resource could be deployed in one and only one resource group. (40) (41, n.d.)

When defining the resource group, below points should be considered:

- All the resources in one group should share the same lifecycle. If one resourced needs to have a different deployment cycle it should move to another resource group.
- Each resource can be included in only one resource group.
- Resources could be added or removed from a resource group at any time.
- It would be possible to move a resource from one group to another one.
- Resources from different regions could be located in one resource group.
- Administrative actions could be limited to each resource group.
- A resource from one group could have a connection with another resource of another resource group with a different life cycle. (For example, web apps connecting to a database).

In order to create a resource group, we need to provide a location where the metadata of the resource group should be stored. If the resource group's region is temporarily unavailable, we can't update resources in the resource group because the metadata is unavailable. (41)

3.2 Region

A region is a set of datacenters deployed within a latency-defined perimeter and connected through a dedicated regional low-latency network. (42)

3.3 Portal

The Azure Portal is actually a web application that could be used for creating and managing resources. It is also be sued for billing information. (40)

3.4 Automation

Consistent management across Azure and none Azure environment would be possible through Automation. (40)

3.4.1 Azure Automation Tools

Azure provides several ways of interacting and automating things. PowerShell and command-Line interface are two main ways for this purpose. PowerShell provides cmdlets for managing services and resources through API. PowerShell cmdlets are used to handle account management and environment like creating, updating and deleting resources. Azure PowerShell is an open source and maintained by Microsoft. Command-Line is a tool to update, create and maintain resources through command-Line. Actually, this tool is created for administrators and operators who are not familiar with Microsoft technology but have experience with Linux or Unix.

3.5 REST API

The common usage of REST API is for the web because the developers do not need to install libraries or additional software. REST has the ability to handle different types of calls, return different data formats and even change structure with the correct implementation of hypermedia. All Services in Azure such as the management portal has specific REST API for their functionality. In this case, they could be accessed by any application which is compatible with RESTful services. (Modi, Damaschke, Klaffenbach, & Michalski, 2018) (43)

3.6 Azure Resources Managers and Tools

The Azure platform consists of three parts; Azure Execution Model which is the areas where services and applications are provided in the cloud, Azure Application Building Blocks and Azure Data Services, which are the services that extend the platform to common capabilities and functionalities. (Modi, Damaschke, Klaffenbach, & Michalski, 2018)

3.6.1 Azure Resource Manager

Azure Resource Manager is the deployment and management service for Azure. It provides a management layer that we can create, update, and delete resources in our account. we can use management features, like access control, locks, and tags, to secure and organize the resources after deployment. (44)

In the following, the benefit of using Azure resources managers have been listed;

- We are able to manage our infrastructure through declarative template rather than scripts. Deploy, Manage and monitor all resources as a group instead of individually
- The solution would be able to redeploy through development lifecycle.

- There is a dependency between resources so they would be deployed in order.
- Access control could be applied to all resources
- Tags could be applied to all resources in order to organize logically. (44)

3.6.2 Consistent Management Layer

When Azure Resource manager receives any request through any of Azure tools from end-users, first of all, authenticate it then sends it to proper Azure service. Since all the requests are handled through the same APIs, the consistent result would be seen in all different tools. (44, n.d.)

The following image shows the role that Azure Resource Manager plays in handling Azure requests;

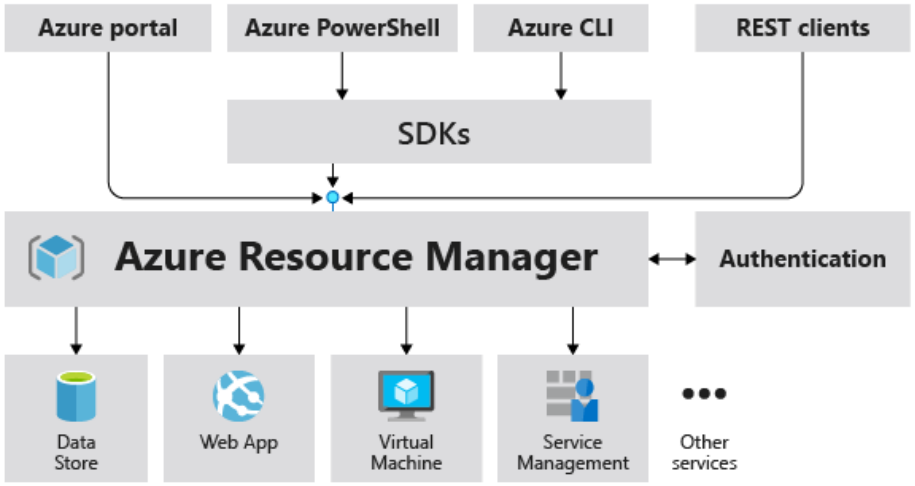

Figure 3. 1

3.6.3 Scope

Azure provides four levels of scope: management groups, subscriptions, resource groups, and resources. The following image shows an example of these layers

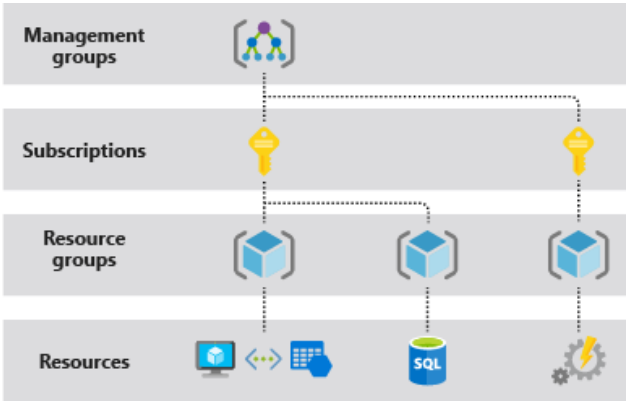

Figure 3. 2

Lower levels inherit settings from higher levels. When a policy is applied to the subscription, the policy is applied to all resource groups and resources as well. However, another resource group doesn't have that policy assignment. (44)

3.6.4 Create a resource group

In order to create a resource group, we should go to Dashboard then select the resource group. After that, click on “Add” button and type a name for the resource group.

3.6.5 Adding a resource to a resource group

In order to add a resource like storage to the resource group, first of all, we need to click on the resource group from Dashboard. Then through “Add” button on the top page, select the source.

3.6.6 Tagging

By tagging resources, we are able to organize them logically. Each tag consists of a name and a value pair. After applying tags, we can retrieve all the resources in our subscription with that tag name and value. By tagging, we are able to retrieve related resources from different resource groups. This approach is helpful when we need to organize resources for billing or management. We need to consider below points while tagging the resources;

- Not all resource types support tags.
- The maximum tag for each resource or group is 50, if we want to use more, we do not have other solution except using JSON string.
- The number of characters of name is limited to 512 characters.
- Tags could not be supported by Generalized VMs.
- The resources of a resource group cannot get tags which applied to the resource group.
- Classic resources such as Cloud Services could not have tags.
- These characters: <, >, %, &, \, ? could not be used in tags.(45)

3.6.7 Tagging resource and source group

In order to add a tag to a source group, first of all, the source group should be selected through Dashboard. Then from left panel, “Tag” should be clicked and add name and value.

The same procedure should be followed to add a tag to a resource.

3.6.8 Resource Locks

By locking resources, we could prevent from accidentally deleting or modifying essential resources by other users in the organization. There are two different levels in portal for locking resources;

- CanNotDelete: it means the authorized users could read and modify resources, but they do not have permission to delete.
- ReadOnly: means authorized users can only read the resources but they do not have permission to update or modify it. (46)

3.6.9 Lock a resource group

In order to lock a resource group or a resource, first of all, we should select a resource group or a resource from Dashboard. Then click on “Lock” option from the left panel and create a lock key the same as the below capture,

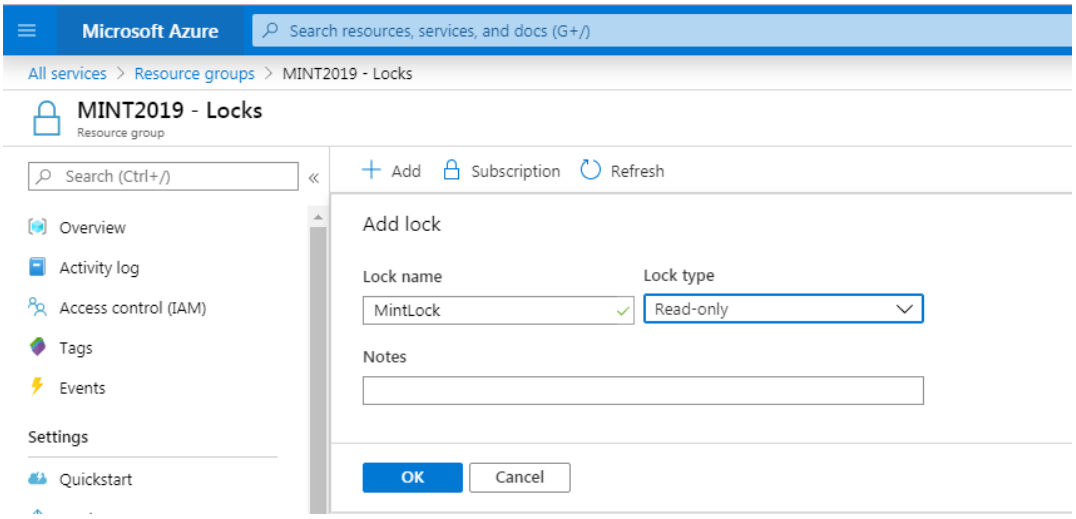

Figure 3. 3

3.6.10 Resource Manager Template

There is a possibility in Azure to convert the whole project as a code. So, everyone would be able to run those code and can have the same environment. In order to get this purpose, we need to use Azure resource manager template. Actually, this template is a JSON file which specifies infrastructure and configuration for the project. In the following, we can see some advantages of using a template;

- Declarative syntax: This feature allows us to deploy an entire infrastructure not only virtual machine but also network devices, storage and other resources.
- Repeatable results: By using template, we could be able to deploy the same template many times and get the same resource types in the same state.
- Orchestration: This feature keeps the priority and order of creating resources. So, we do not need to worry about the complexities of ordering operations.

- Built-in validation: First of all, a template is checked for validation then it starts to deploy.
- Modular files: It would be possible to break a template into smaller components. We also be able to run one template inside the another one.
- Create any Azure resource: Whenever we would like to use new services in our template, we would be able to add it to our template.
- Tracked deployments: Through Azure portal, we would be able to track the history of our template such as the parameters which have been passed in and any output value. (47)

3.7 Active Directory

Azure AD could be integrated to Local AD servers in the organizations in order to manage access to cloud-based SaaS applications. An organization is also able to easily implement Single Sign-On on a multi-factor authentication through Azure AD without adding third party software into their environment. Azure AD is a multi-tenant, Geo-distributed, high availability service running in every Microsoft data center around the world. Microsoft has implemented automated failover with a minimum of two copies of your Azure Directory Service in other regional or global data centers. The main directory is running in the main primary data center, is regularly replicated into another two in your region. If we only have two Azure data centers in your region, as in Europe, the copy will distribute to another data center in another region; (Modi, Damaschke, Klaffenbach, & Michalski, 2018)

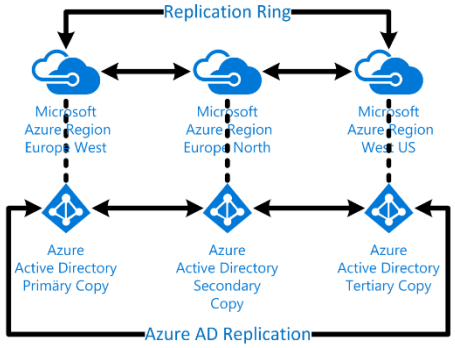

Figure 3. 4

There are four selectable options for Azure Active Directory with different features.

AD free:

- The number of objects would go up to 5,00,000 which they consist of users, devices, applications, or groups.

- User or group management including adding, updating and deleting.
- Providing SSO for ten applications per each user.
- Connecting and updating with local AD.
- Providing three basic security and usage reports.

AD Basic: This option would have all the advantages of free AD in addition below features.

- Self-service password reset for the cloud users.
- Company branding (logon pages / access panel customization).
- Application proxy.
- Service level agreement 99.9%.

AD Premium P1: This Supports common features from the basic Azure AD and more.

- Self-service group and app management / self-service application additions / dynamic groups.
- Self-service password reset/change/unlock with on-premises write-back.
- Multi-factor authentication (cloud and on-premises (MFA server)).
- MIM CAL plus MIM server.
- Cloud app discovery.
- Connect health.
- Automatic password rollover for group accounts.

AD Premium P2: This includes all the capabilities in Azure AD Premium P1 as well as the new Identity Protection and Privileged Identity Management capabilities. (Modi, Damaschke, Klaffenbach, & Michalski, 2018)

3.7.1 Deploy Azure AD

It is really important to note that Azure Active Directory is directly connected to your Azure subscription. So, there can only be one account administrator per each Azure subscription. The account administrator is the only one who can manage Azure AD and subscription connections. If we lose your administrator credentials or lose access to the administrator account, we can no longer manage our subscription. To create a subscription, the subscription administrator must have a Microsoft Accounts or former named Live ID or Microsoft Account, Azure Active Directory account. Normally an Azure Active Directory is created when we create an Azure subscription, or we subscribe to a Microsoft Cloud service like Office 365. As an Azure account

administrator, we can create a new Azure AD and change our Azure subscription to the new Azure AD. (Modi, Damaschke, Klaffenbach, & Michalski, 2018)

In order to deploy an AD, from “Azure Services”, Azure Active directory should be selected.

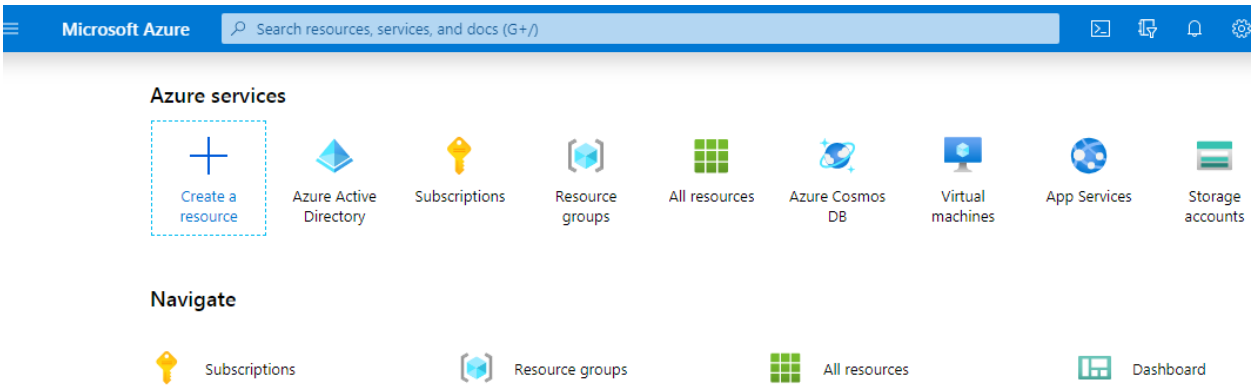

Figure 3. 5

Then from “create a directory” button, and filled out required information, a new AD is created.

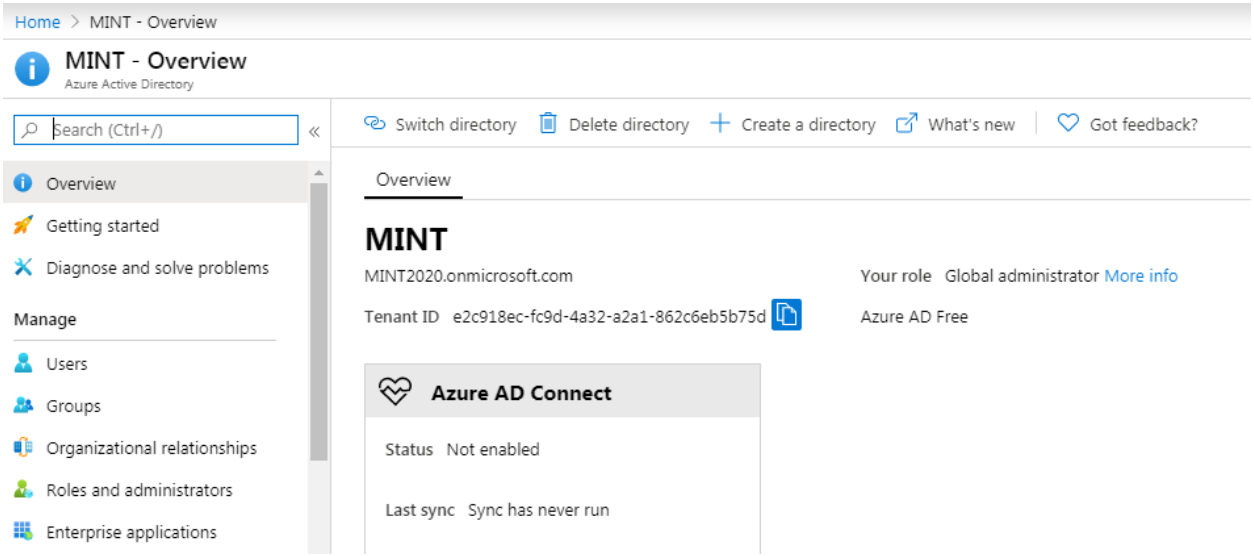

Figure 3. 6

3.7.2 Adding account and groups to Azure AD

Basically, we have two types account;

- Cloud accounts: Accounts that are created via Azure Active Directory or other Microsoft Cloud services, like Office 365.
- Hybrid accounts: Accounts that are created and located in on-premises Microsoft Active Directory Domain Services. Those accounts are deployed via the Azure Active Directory connect and synchronization tool. (Modi, Damaschke, Klaffenbach, & Michalski, 2018)

In order to create a cloud account, first, we need to select the Active directory then from the left panel click on the “Users” and create a new user by using “New User” button on the top. Then provide the username, name and other required information and click on the “Create” button. The same as the following snapshot

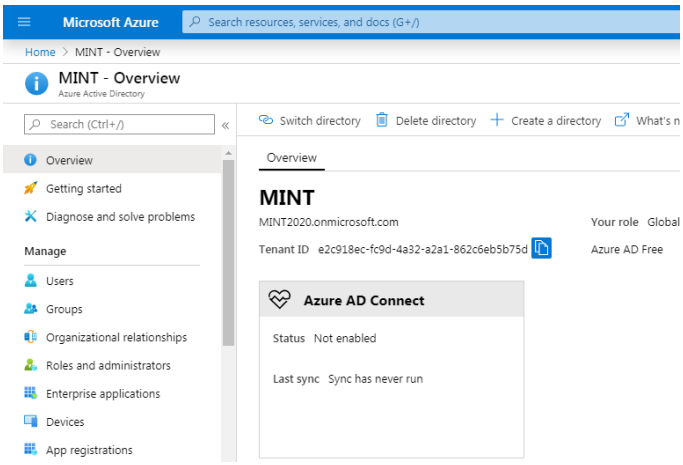

Figure 3. 7

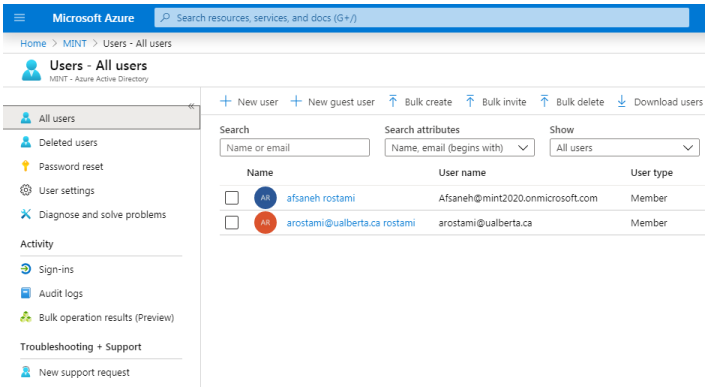

Figure 3. 8

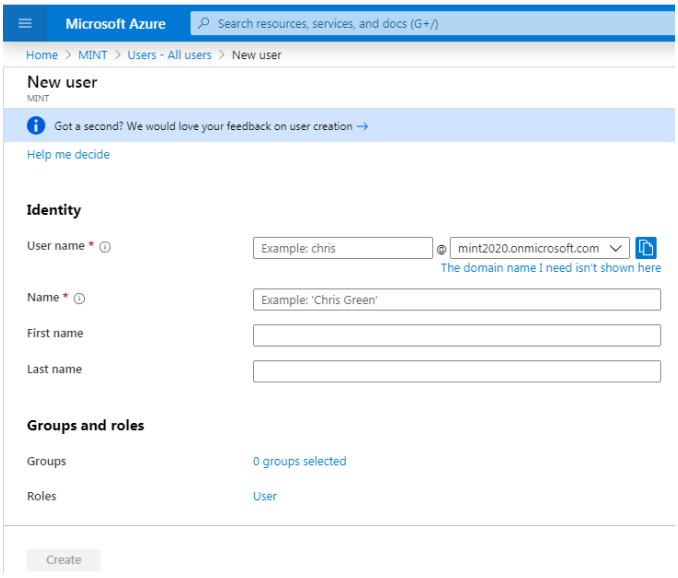

Figure 3. 9

Also, we can assign different roles to users by using "Assign roles" from the left panel and add different kinds of roles to the users;

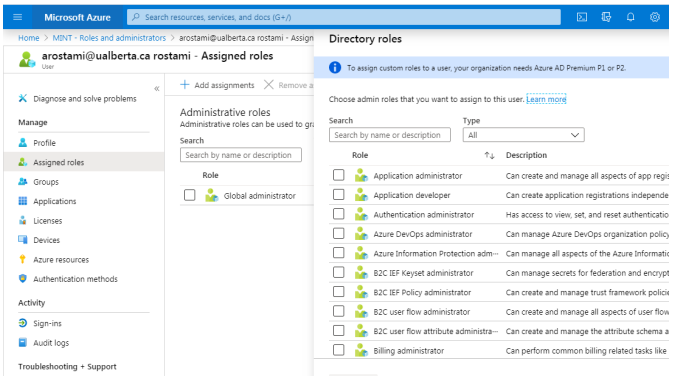

Figure 3. 10

3.8 Azure AD connect

Azure AD Connect is the Microsoft platform designed to meet the hybrid identity objectives and achieve them. The following features could be provided through AD connect;

- Password hash synchronization : A sign-in method that synchronizes a hash of users on Local AD password with Azure AD.
- Pass-through authentication: A sign-in method that users can use the same password on local AD and the cloud.
- Federation integration: Federation is an optional of Azure AD Connect
- Synchronization: It can create users, groups, and other objects. Also, it matches identify information of users and groups between Local AD and the cloud.

- Health Monitoring: Monitoring could be provided through this option. (48)

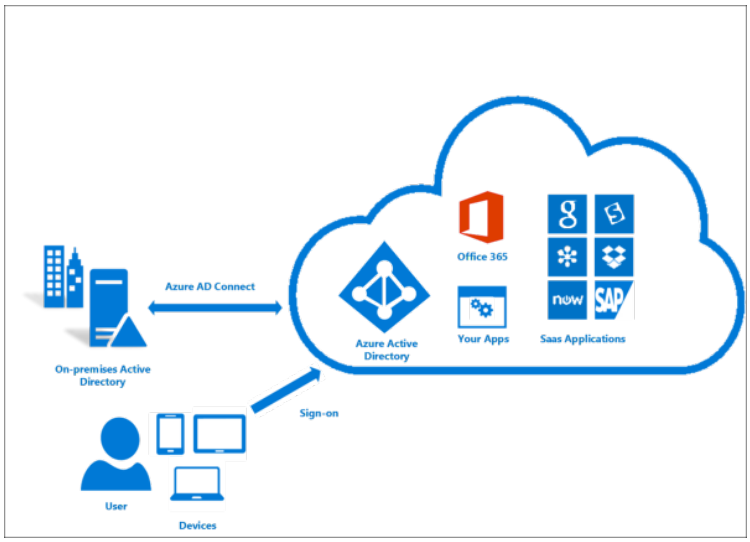

Figure 3. 11

3.8.1 Installing AD Connect

It should be mentioned that Federations provide a standard service that enables safe identity information sharing between trusted business partners across an extranet. For example, when a user from one of the partners in federation needs to have access to a web application which is hosted by another partner, It is the responsibility of the user's own organization to authenticate the user and to provide identity information in the form of claims to the hosting partner. The hosting partner, by using the trust policy, the received claims is mapped to a claim which is understood by the web application. Practically, Azure AD and Microsoft becomes our business partner in AD AFS. The following prerequisites should be provided in advance;

- An Azure subscription in order to access the portal.
- An AD global administrator account for Azure AD that we want to integrate it.
- An AD Server or member server with win server 2008 or newer.

Before installing AD account, two accounts should be created. The first one is a service account with enterprise admin rights. The second one is a global administrator in Azure AD.

These two accounts will perform actions on both directories, so the users on Local AD will not access to Azure AD and vice versa. The following diagram, shows the basic workflow behind the communication between local AD, Azure AD, and AD connect machine. (Modi, Damaschke, Klaffenbach, & Michalski, 2018)

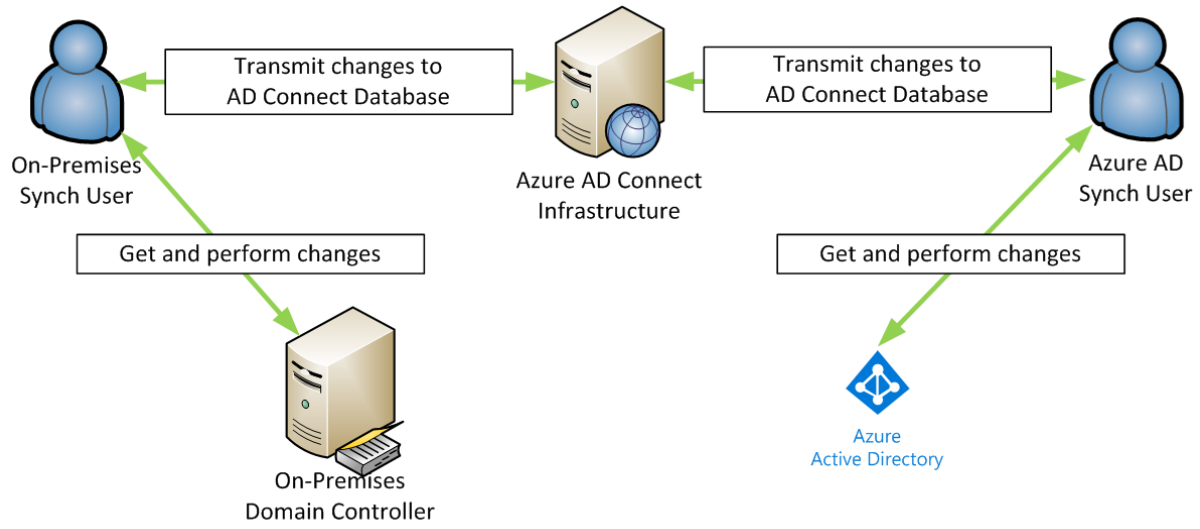

Figure 3. 12

After choosing which type of user sign-on works for our scenario, the wizard changes. In ADFS implementation, we should get more steps to get the federation running. In this way, we need one ADFS server and one ADFS proxy and also an additional service account. The link would be based on windows remote management. (Modi, Damaschke, Klaffenbach, & Michalski, 2018) In the following, all steps for implementation Azure AD with password synchronization have been listed;

- Azure AD connect should be downloaded. Then from the start menu, Azure AD connect should be clicked.
- Accept the license and click on continue.
- Click on the “Customize”.
- Select “Password Synchronization” and click on “Next”.
- Enter the Azure AD credential and click on the “Next”.
- Enter the local administrator account.
- In this step, we need to select the attribute in our Local Domain which identifies UPN (user principal name) in Azure AD.
- We need to select domains and OUs for synchronization.
- We should define how the users should be identified over our domain.
- Select users’ groups which should be synchronized.
- In this step, we choose which features should be synced in addition to the attributes.

- We need to decide which azure AD apps should be synced.
- Afterwards we need to limit the attributes from Active Directory that would be transferred to Azure AD.
- In this step, we need to choose which directory extension attributes from local AD are transferred to Azure AD.
- Click on the “Install” button. (Modi, Damaschke, Klaffenbach, & Michalski, 2018)

3.8.2 Benefit of AD connect

When the local AD and Azure AD is integrated with each other, users are able to use the same identity in order to access both the cloud and Local resources. Users and organizations can take benefit from:

- Users can use the same identity for both the cloud services such as office 365 and local applications.
- By using one tool, this synchronization would be done
- Also, it provides the newest features and there would be no need to use old versions of identity integration tools such as DirSyn and Azure AD Sync. (48)

3.9 AD Connect Health

Azure Active Directory Connect Health enables us to monitor on-premises infrastructure. By using this feature, we can maintain a reliable connection to office 365 and other Microsoft Online Services. Such Durability is accomplished by supplying the main identity elements with monitoring capabilities. It also makes the key data points easily accessible for these components. (48)

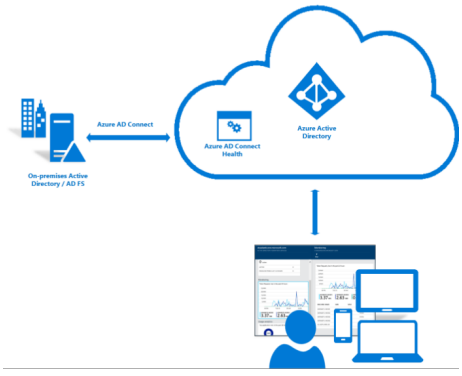

Figure 3. 13

3.9.1 Benefit of AD connect Health

By using Azure AD connect, users become more productive because they can use the same identity to access local resources and cloud services. Azure AD Connect Health helps track and

gain insight into your on-site identity system so that this environment is secure. It is very simple; we just need to install an agent on each Local identity server. At the time of writing this report, AD Connect Health for AD FS supports AD FS 2.0 on win Server 2008R2, Win Server 2012, Win Server 2012R2 and Win Server 2016. It also supports monitoring for AD FS Proxy or web application proxy server that authenticates extranet access. In the following, we can see the key benefits. (48)

Key Benefits	Best Practices
Enhanced security	Extranet lockout trends Failed sign-ins report In privacy compliant
Get alerted on all critical ADFS system issues	Server configuration and availability Performance and connectivity Regular maintenance
Easy to deploy and manage	Quick agent installation Agent auto upgrade to the latest Data available in portal within minutes
Rich usage metrics	Top applications usage Network locations and TCP connection Token requests per server
Great user experience	Dashboard fashion from Azure portal Alerts through emails

Figure 3. 14

3.10 Azure Virtual Network (Vnet)

Azure Virtual network is one of the main building blocks for our private network in Azure. VNet provides several types of Azure resources like Azure Virtual Machines which communicate securely with each other and with internet and with our local Data Center. VNet is similar to our own data center but with additional advantages such as scale, availability and isolation. The following concepts exist in VNet; (49)

- Address Space: By creating a VNet, we need to specify a custom private IP Address range. Azure assigns an IP address from IP address range that have been identified in advance.
- Subnets: Subnets provide us segmentation within virtual network. Then we can deploy Azure resources in a specific subnet. This subnetting also improves IP address allocation efficiency. Moreover, it enables us to secure our resources.

- Regions: VNet is divided into one or multiple locations although multiple virtual networks from different region still communicate with each other by using Virtual Network Peering.
- Subscription: VNet is divided into a subscription. We can deploy several virtual networks within each Azure subscription and region. (49)

While we are designing a VNet, we need to pay attention to below points;

- Ensure non-overlapping address space: our VNet address space should not have any overlapping with our organization's network range.
- Regarding subnetting, it should not cover the entire address space, we need to keep some addresses for future.
- It would be recommended to design some large VNet than multiple small VNet because of management overhead.
- VNet should be secure by Network security Group. (49)

3.11 Security Groups

By using Security Groups, we can make limitations for inbound and outbound traffic. Security groups contain security rules that allow or deny network packets from several types of Azure resources. Each rule in security groups contains following properties:

- Name: A unique name should be assigned to each rule within a security group.
- Priority: A number between 100 to 4096 which identifies the order of processing. Rules with lower number would be processed before high numbers.
- Source/ Destination: Network security groups are processed after translating public IP address to Private for inbound traffic and before translating private IP address to public one for outbound traffic. By specifying a range IP address, service tag or application security groups, we can create fewer security rules. Augmented Security rules are those rules that we can specify a range IP address or multiple individual IP addresses.
- Protocol: TCP, UDP, ICMP or Any
- Direction: It could be inbound or outbound
- Port Range: We can specify an individual or a range of ports
- Action: It could be Allow or Deny. (50)

3.12 Augmented Security Rules

By Using Augmented security rules, we can create large and complex network security policies with fewer rules. It is possible to combine multiple ports and multiple explicit IP addresses and ranges into a single security rule. In order to make simple maintenance of security rules, we can combine segmented security rules with service tag or application security groups. (50)

3.13 Service Tag

A service tag is a group of prefixes of IP addresses from a given Azure service. It is useful to minimize the complexity of frequent updates to network security rules. We can use service tag to specify network access controls on network security rules or Azure Firewall. (51)

3.14 Application Security groups

This feature provides a possibility to configure a network security as a natural extension of an application structure. We can make a group of VMs and specify a security group based on this VMs group. By constantly managing specific IP addresses, you can reuse your security policy on a scale. This platform maintains the complexity of explicit IP address and multiple rule sets that enable us to focus on our business logic. In order to make it clear, consider the below example.

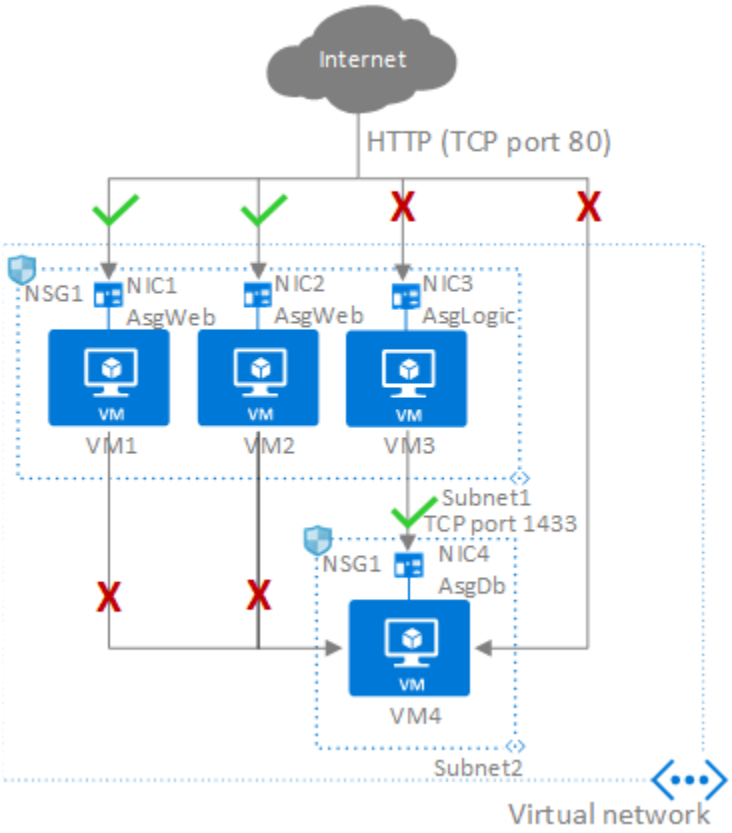

Figure 3. 15

In this example, NIC1 and NIC2 are members of “AsgWeb” application security group. NIC3 is a member of “AsgLogic” application security group and NIC4 is a member of “AsgDb”. The following rules have been assigned to NSG1;

- All inbound internet traffic with destination port 80, Protocol TCP and destination application “AsgWeb” have been allowed.
- All database traffic with destination “AsgDb” and destination port 1433 with any protocol has been denied.
- All Database-BusinessLogic with source “AsgLogic” and Destination “AsgDb”, destination port 1433 by Protocol TCP has been allowed.

The rules specify an application security groups as the source or destination only applied to those network interfaces that are a member of the application security group.

Applications security groups are subject to the following restrictions:

- The number of application security groups is pre-defined per as a subscription.
- We can specify only one application in source or destination not multiple.
- It is not possible to add network interface from different Vnets to the application group policy.
- Applications which used in source and destination in a security rules, should be exist in the same Vnet. (50)

3.15 Communicate with internet

All resources in a VNet can communicate to the internet by default. In order to provide access to resources from the internet, we need to use a public IP address or a public load balancer. (49)

3.16 Load Balancer

Load balancing refers to spreading load or incoming traffic across a group of resources or servers. (52)

Azure Load balancer is a layer 4 (TCP, UDP) of OSI Model that is the single point of contact for clients. It distributes all incoming traffic of front-end interface to back-end pool instances according to specific rules and health probe. The back-end pool instance could be Azure Virtual machines. With Load balancer, we can scale up our applications and create high available services. It also supports both inbound and outbound traffic with low latency and high throughput. (52)

A public load balancer is used for outbound traffic which translates the public IP address to a private and distributes the internet traffic over inside VMs. (52)

An internal or private Load balancer is used for scenario where only private IP addresses are needed. (52)

The components of load balancers have been listed at below;

- Frontend IP configuration: This is an IP address of load balancer which is the point of contact for clients. It can be Public or private which either creates a public or internal load balancer
- Backend Pool: The group of VMs or instances that are going to serve incoming traffic. In order to meet high volume of incoming traffic, we need to increase the number of instances in the backend pool. Load balancers constantly reconfigure themselves via automatic reconfiguration when the number of instances in backend pool change and there is no need for any additional operations for this case.
- Health Probe: This is used to determine the health of an instance in the backend pool. When an instance fails to respond, the load balancer stops sending traffic to that instance. Load Balancers provide different health probe type for TCP, HTTP and HTTPS endpoint.
- Load Balancing Rules: That is a rule which tells lad balancer when and what action should be done.
- Inbound NAT rules: This rule forwards traffic from a specific port of IP address to a specific port of a specific backend instance. It is possible to map multiple internal endpoints to ports on the same front-end IP address.
- Outbound rules: An outbound rule map some instances of the backed pool to the frontend. (52)

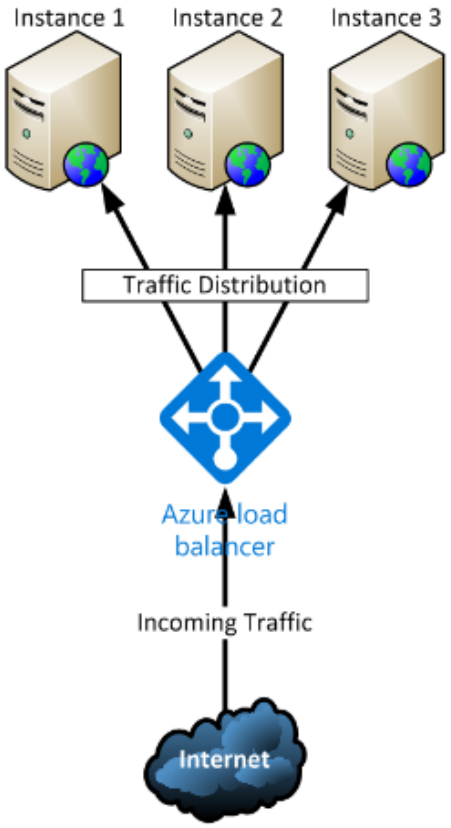

Figure 3. 16

3.17 Load Balancer Concept

Load balancers provide come capabilities for TCP and UDP applications;

- Load balancing algorithm: Load balancers use a hashing algorithm for distribution inbound traffic and rewrite the header of received packets to backend pool instances. By default, load balancers use a 5-tuple hash that incudes source IP, source port, destination IP, destination port and protocol number. It is possible to create a hash by using 2 or 3 tuple hash. (52)

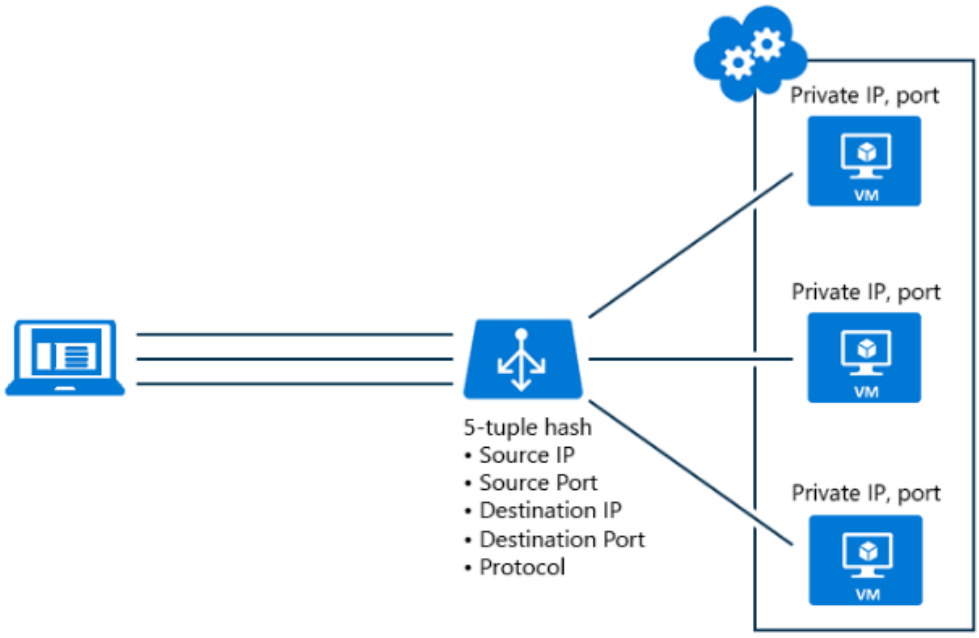

Figure 3. 17

Application independence and transparency: Load balancers will not directly interact with TCP or UDP or application layer. It also will not terminate or originate any flow of traffic. Always between a client and a back-end pool instance protocol hand shaking has occurred. A response to an inbound traffic is from a virtual machine and once a flow arrives on the virtual machine, the original source IP address is kept. Only a VM responds to each end point. For more detail, it should be mentioned that a response to a request of a front-end interface has been made by an instance in a back-end pool. When we are checking connectivity to a front-end, practically we are checking connectivity to at least one instance in a backend. Any UPD and TCP could be supported by Load Balancer. Because Load Balancers will not interact directly with TCP Payload and provide TLS connections offload, we would be able to make end-to-end encrypted connections. Because Load Balancers terminate TLS Connection on the VM, we would get advantages of large scale-out

- Outbound Connections (SNAT): All outbound traffic from private IP addresses inside our VM to a public IP address on the internet can be translated by Load Balancers. Once a public front-end would be mapped to a backend VM by using a load-balancing rule, indeed Load balancer translate outbound traffic to Public IP address of front-end interface. The advantages of this configuration are;

Because front-end could be mapped to another instance of a service, we could easily upgrade or make disaster recovery.

It would make ACL Management so much easier. Translating outbound links to fewer IP addresses than the computer eliminates the cost of enforcing secure lists of recipients. (52)

3.18 Hash-Based Distribution

By default, this type of Load balancers uses a 5-tuple hash to map traffic to available services and servers. It only offers stickiness in a transport session. Packets in the same TCP or UDP session are sent to the same load balancer instance. When the sender site is closed and the session is responded or the new session begins from the same source IP, the source port changes. This can result in the traffic being diverted to a different data center and the source of the load balancer. (Modi, Damaschke, Klaffenbach, & Michalski, 2018)

3.19 Port Forwarding

Load balancers enable us on how to manage inbound traffic. These kinds of traffic can come from the internet or virtual machines in other cloud services or virtual networks. Load balancer listens on a public port and forward traffic to an internal port. This internal port could be the same for internal and external sources or could be different. By using port forwarding, we can redirect traffic from an incoming port to another port that our server is listening. (Modi, Damaschke, Klaffenbach, & Michalski, 2018)

3.20 Automatic Reconfiguration

The Load balancer changes their configuration all the time when we change the number of instances in their sites. For instance, when we add or delete a new server or instance. (Modi, Damaschke, Klaffenbach, & Michalski, 2018)

3.21 Service Monitoring

The load balancer will check the safety of the different instances of the servers. For instance, when a server fails to respond, the load balancer stops sending new traffic to that server. Three types of service monitoring exist;

- Guest Agent probe (on PaaS VMs only): The load balancer in the virtual machine uses the Azure guest agent. The guest agent listens and responds via HTTP port 200.If the agent fails to respond, the load balancer marks that instance as an unresponsive and stops sending traffic to that instance but constantly checks the status of the agent by ping until it responds or removes from load balancer set.

- HTTP custom probe: This type overwrites guest agent. It provides to custom the logic in which the health of instance is determined.
- TCP Custom probe: This type trusts on the TCP session to a defined probe port. (Modi, Damaschke, Klaffenbach, & Michalski, 2018)

3.22 Communication between Azure resources

Azure resources could communicate with each other through one of the below ways;

- A virtual network: When we deploy a VNet and some Azure resources inside the VNet, they communicate with each other without any extra configuration. (49)
- A virtual network service endpoint: It is possible to extend the virtual network's private address space and the virtual network identity over a direct connection to Azure services such as Azure storage account and Azure SQL Database. (49)
- VNet Peering: By using this option, we would be able to link two virtual networks in the same region via using Private IP address. Indeed, VNet peering routes traffic between virtual networks through the internal Azure backbone network. There would not be any gateway between these two virtual networks. This connection would be low-latency, high-bandwidth link. Another advantage of this option is that the resources from one virtual network could be used by another virtual network. For instance, the gateway from one of virtual networks could be used by another one. (Modi, Damaschke, Klaffenbach, & Michalski, 2018)

3.23 Communicate with on-premises resources

We would be able to make a connection between our local resources and virtual network in Azure by using one of the following ways:

- Point-to-site (P2S) virtual private network (VPN): this connection makes us able to create a secure connection between clients' computers and our virtual networks. This solution would be useful to connect to the virtual network from a remote location. Also, it

does not need to use any public IP address in client computers. (53)

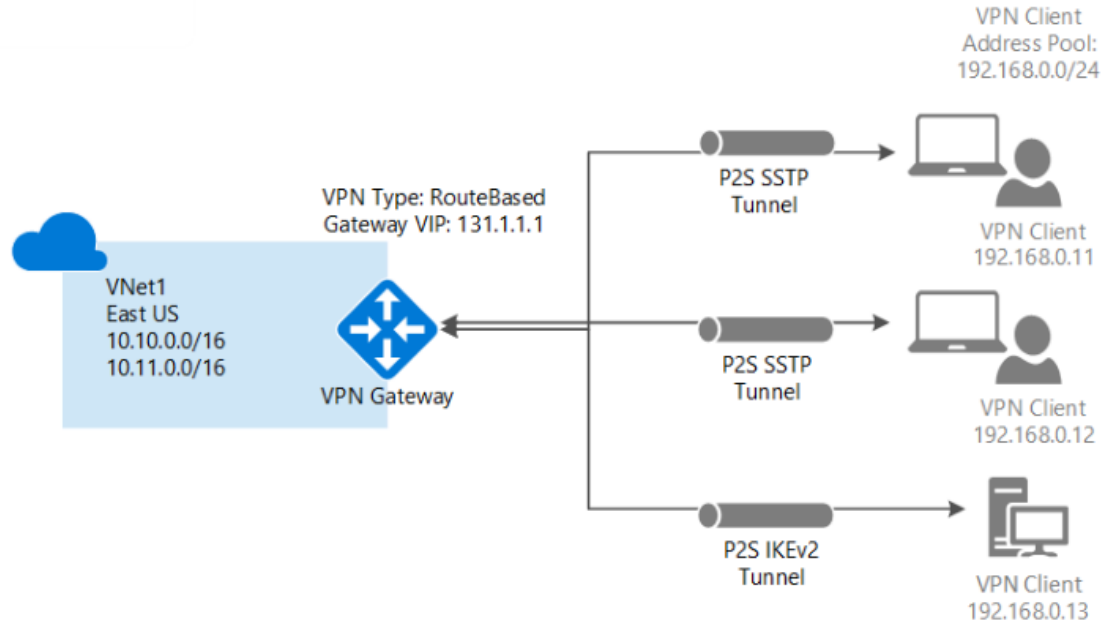

Figure 3. 18

- Site-to-Site VPN: This type of connections is called S2S VPN Gateway. Indeed, it is a gateway connection over IPsec/IKE VPN tunnel which needs to setup a VPN device on-premises site with one Public IP address. Also, this connection is used for a cross-premises and hybrid configuration. (53)

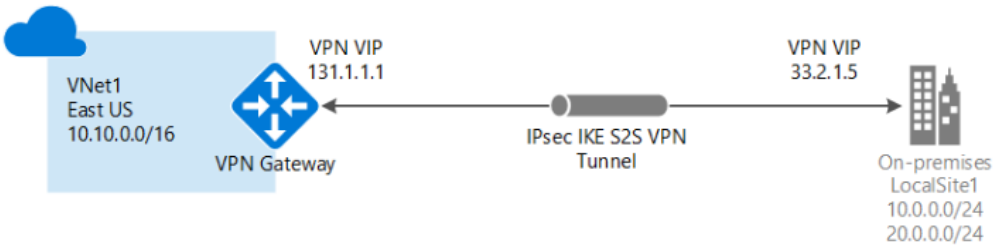

Figure 3. 19

- Azure ExpressRoute: This type of connection extends our local network into Azure cloud via a private connection. This connection could make from any IP VPN network, point-to-point Ethernet network or even virtual connection to Azure cloud services such as Microsoft Azure, Office 365 and even CRM online tools. One of the main features of this connection is that it does not go through Public internet. This connection provides a link with faster speed, lower latencies and higher security. A virtual network gateway is required for this connection and it should be configured as “ExpressRoute” rather than

VPN. By default, traffic over ExpressRoute is not encrypted but we would be able to encrypt it as well. (53, n.d.)

3.24 Filter network traffic

We would be able to filter traffic between subletting by using either or both following option:

- Security Groups: Network security groups and application groups could contain multiple inbound and outbound security rules which enables us to filter traffic from or to resources by using source IP, destination IP, Protocol, and port number. (53)
- Network virtual appliance: A network virtual appliance is a virtual machine that does some network functions such as firewall, WAN optimization and other network operation. (53)

3.25 Route Network Traffic

By default, Azure routes traffic among connected virtual networks, local networks, subnets and Internet. By using either or both the following ways, we could be able to override the default route in Azure; (49)

- Route table: We would be able to create or customize Azure's default route or add additional routes to a subnet's route table. First, we need to create a route table then associate it with zero or more virtual network subnets. Also, each subnet could have zero or only one route table that associated to it. Regarding the maximum number of routes, we need to check the limitation of our subscription. When we create a route table and associate it to a subnet, all the routes would be combined with default route by default. To specify next hop types in user-defined route, we would be able to use the below options. (54)
- Virtual Appliance: Virtual appliance is a VM that could perform some of network operations the same as a firewall. The IP address for a virtual appliance could be Private or Public address.
- Virtual network gateway: We would be able to specify a flow with a specific address range to route to a virtual network gateway. In this case, virtual network gateway should be configured as VPN not ExpressRoute. (55)
- None: It would be used when we want to drop a traffic flow rather than routing the flow to a specific destination. (55)

- Virtual network: It is used when we want to over-ride the default routing within a virtual network. (55)
- Internet: It is used when we want to route a traffic to an address in Internet or we want to keep traffic within Azure backbone network while the destination of this traffic is Azure services with Public IP address. (55)
- BGP Routes: If we want to make a link between our local network and Azure VPN Gateway or ExpressRoute, we could propagate our Local network to Azure virtual network by using BGP Roués. (49)

3.26 Azure VPN Gateway

Indeed, VPN gateways are as core routers and firewalls inside our Azure networks. They could provide below features; (Modi, Damaschke, Klaffenbach, & Michalski, 2018)

- Internet gateway
- Site-to-Site VPN gateway
- Point-to-site VPN gateway
- Express Route gateway
- Vnet-to-Vnet gateway

Each VNet could have at least one VPN gateway. VPN gateways are available with different features and services; (Modi, Damaschke, Klaffenbach, & Michalski, 2018)

	VPN gateway throughput	VPN gateway max IPSEC tunnels	Active - Active VPN	ExpressRoute gateway throughput	VPN gateway and ExpressRoute coexist
Basic	100 Mbps	10	No	No	No
Standard	100 Mbps	10	No	1000 Mbps	Yes
High Performance	200 Mbps	30	Yes	2000 Mbps up to 10000 Mbps	Yes

Figure 3. 20

(Modi, Damaschke, Klaffenbach, & Michalski, 2018)

The below diagram illustrates on how using basic VPN gateway, we could connect Azure network to local Network; (Modi, Damaschke, Klaffenbach, & Michalski, 2018)

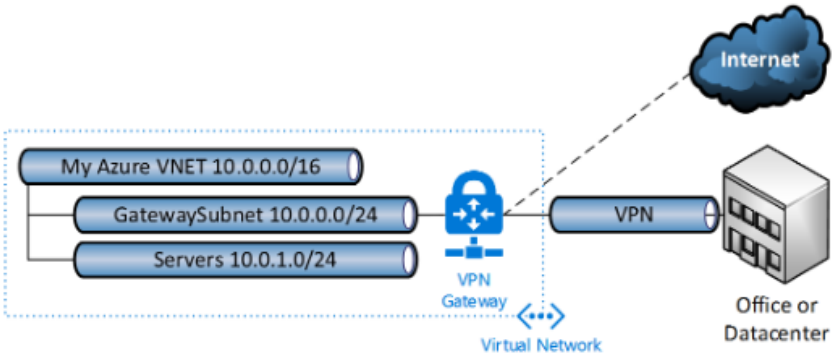

Figure 3. 21

(Modi, Damaschke, Klaffenbach, & Michalski, 2018)

With using standard or performance gateway, the connection would be like the below one;

(Modi, Damaschke, Klaffenbach, & Michalski, 2018)

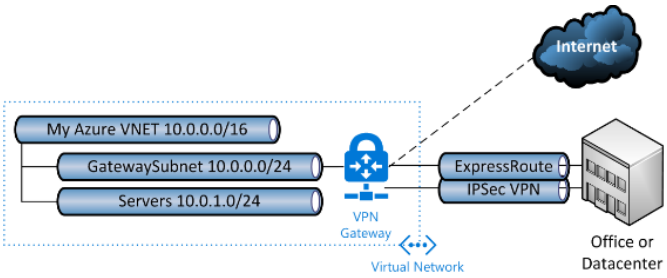

Figure 3. 22

When we want to setup a gateway, we need to determine which kind of gateway would be meet our requirement. Depending on WAN solution, we need to deploy RouteExpress or VPN. Regarding VPN we need to choose Route-Base or Policy-Base. It means that we should determine that we need to enable dynamic routing or static routing with using IPSEC IKE. In the following, there is some more explanation which would be useful to choose VPN type; (Modi, Damaschke, Klaffenbach, & Michalski, 2018)

	Policy-based basic VPN gateway	Route-based basic VPN gateway	Route-based standard VPN gateway	Route-based high performance VPN gateway
Site-to-site connectivity (S2S)	Policy-based VPN configuration	Route-based VPN configuration	Route-based VPN configuration	Route-based VPN configuration
Point-to-site connectivity (P2S)	Not supported	Supported (can coexist with S2S)	Supported (can coexist with S2S)	Supported (can coexist with S2S)
Authentication method	Pre-shared key	Pre-shared key for S2S connectivity, certificates for P2S connectivity	Pre-shared key for S2S connectivity, certificates for P2S connectivity	Pre-shared key for S2S connectivity, certificates for P2S connectivity
Maximum number of S2S connections	1	10	10	30
Maximum number of P2S connections	Not supported	128	128	128
Active routing support	Not supported	Not supported	Supported	Supported

Figure 3. 23

(Modi, Damaschke, Klaffenbach, & Michalski, 2018)

In the following, diagrams illustrate some more explanations about gateway configuration;

- Policy-based VPN gateway should only use when there is no other option. These days most firewalls are able to work with route-based VPN, If there is not this capability on the firewall, we could switch to a virtual network device; (Modi, Damaschke, Klaffenbach, & Michalski, 2018)

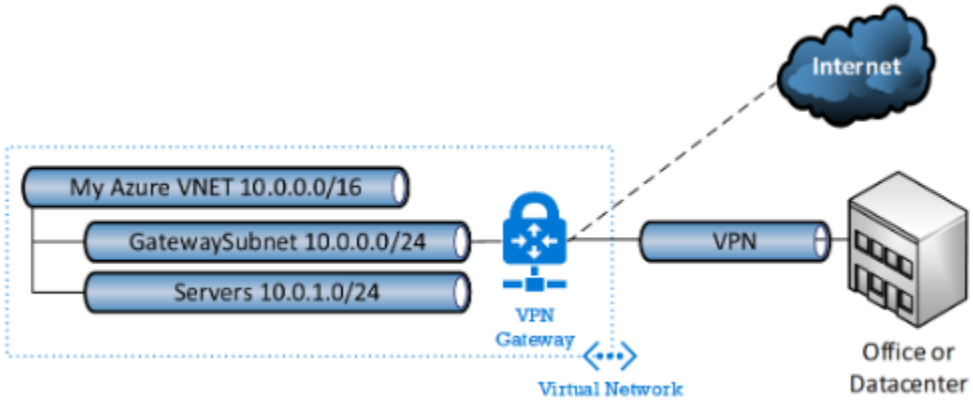

Figure 3. 24

(Modi, Damaschke, Klaffenbach, & Michalski, 2018)

- Route-based VPN gateway with ExpressRoute has been illustrated by the below figure;

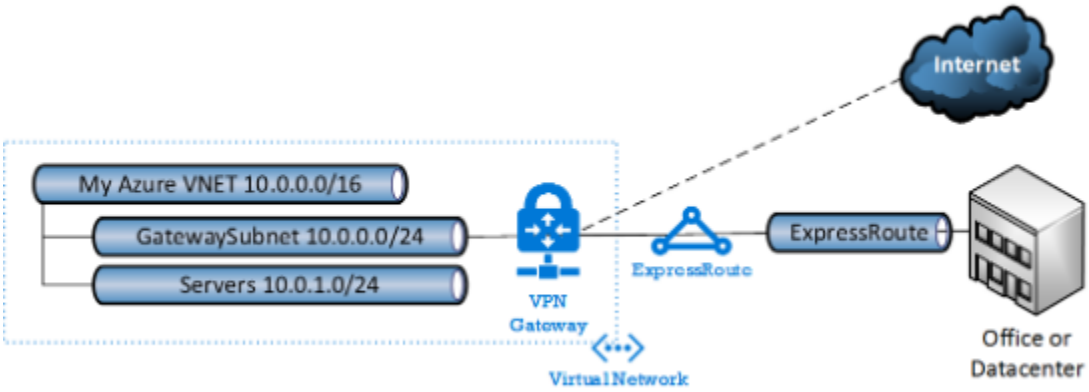

Figure 3. 25

(Modi, Damaschke, Klaffenbach, & Michalski, 2018)

- In the following, it shows the basic structure for VPN gateway with a site-to-site VPN or point 2 site VPN;

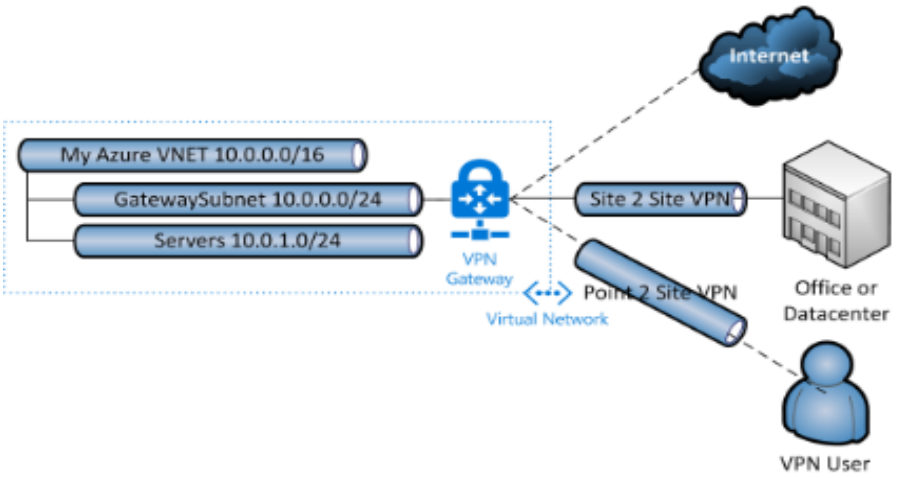

Figure 3. 26

(Modi, Damaschke, Klaffenbach, & Michalski, 2018)

- Route-based VPN gateway with site-to-site or ExpressRoute;

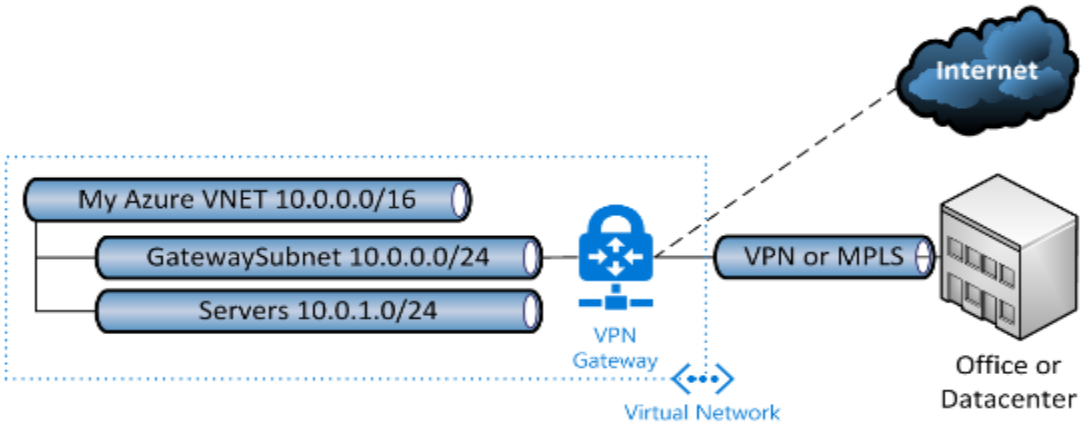

Figure 3. 27

(Modi, Damaschke, Klaffenbach, & Michalski, 2018)

- Route-based VPN gateway with site-to-site VPN and ExpressRoute;

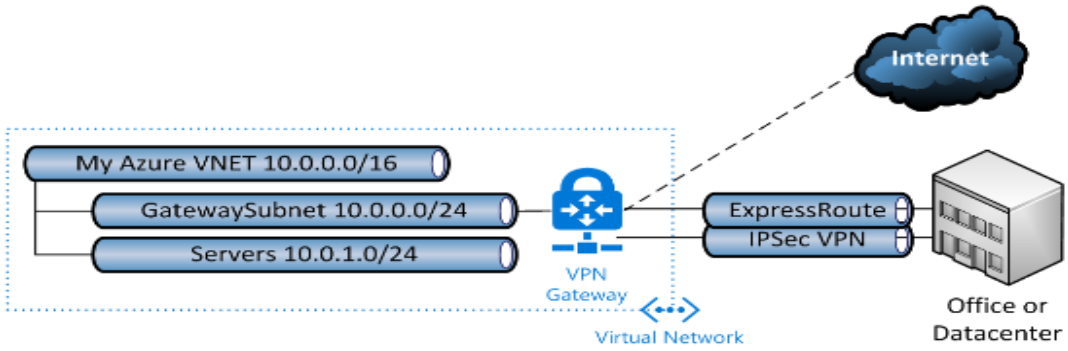

Figure 3. 28

3.27 BGP with VPN Gateway:

BGP is the only the protocol which is used in the internet to exchange routing information between two or more networks. By using BGP, Azure VPN gateway and our local VPN device would be peers or neighbors in order to exchange the route information. Also, it would be possible to make a connection link between local network and Azure virtual network by using static routes instead of using BGP, but BGP provide us the below advantages; (56)

- Support automatic and flexible prefix updates: With using BGP, we only need to specify a minimum prefix to a predetermined BGP peer over IPsec S2S VPN tunnel. Then we would be able to control which Local prefixes could be advertised to Azure virtual network via BGP. Also, it would be possible to advertise large prefixes as well that would be included VNet address prefixes. (56)
- Support multiple tunnels between a VNet and Local site with automatic failover based on BGP: It would be possible to make several connections between Azure networks and our Local network. This capability provides several paths between two connection and all those connections are in active-active mode. If one of the tunnels goes down, BGP routes all traffic to the other tunnel which is in good status. The below diagram shows this structure; (56)

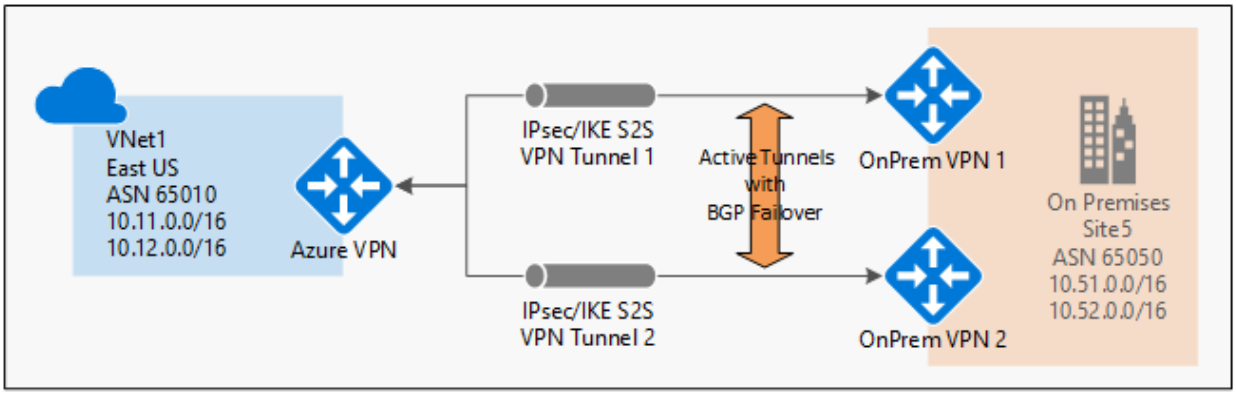

Figure 3. 29

- Support transit routing between your local network and multiple Azure Vnets: BGP brings the capability for gateways to learn prefixes and propagate from different networks. This feature enables VPN gateway to do transit routing between Local sites or across several Azure networks. The following diagram illustrates an instance of a multi-hop topology with several paths that transit routing could be done between two local networks via Azure VPN Gateway. (56)

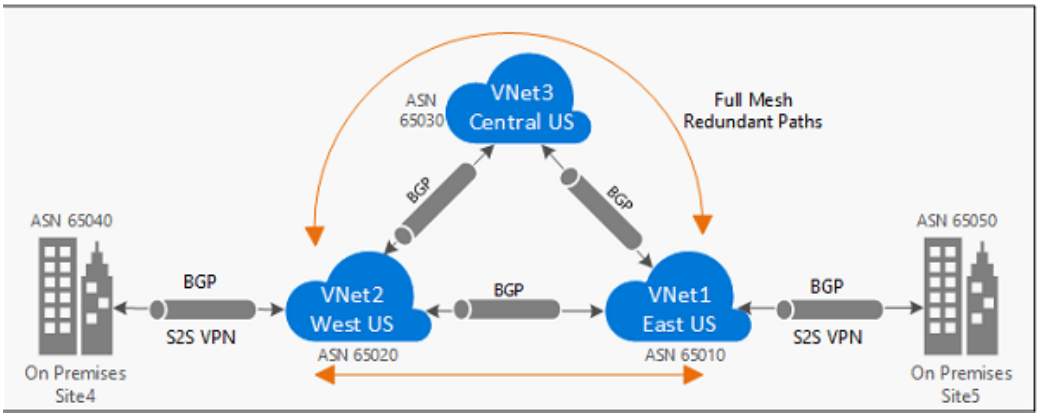

Figure 3. 30

3.28 Storage service

There are two kinds of storage services in Azure;

- Standard storage which includes Blob, table, Queue and file storage account.
- Premium storage: Azure VM disks. (58)

3.28.1 Standard storage service

With using standard storage account, we could have access to Blob storage, Table storage, File storage and Queue storage which are explained at the following; (58, n.d.)

- Azure Blob storage: This storage would be useful to store unstructured data that includes pictures, videos, music, documents and row data. These kinds of data would be stored in directory like container the same as AWS S3. It would be possible to store any number of blobs files up to 500TB. Moreover, security policy could be applied to this type of account. This storage could be used for data backup as well. (58)
  This kind of service comes with three types;
  Block Blob: This type would be useful for storing image, video and documents.
  Append Blobs: That is similar to Block Blob but also is used for appending operation like logging.
  Page Blob: This type would be useful for objects which needs frequent read-write. Also, it used in Azure VMs to store OS and data disks. (58, n.d.)
- Azure Table storage: As its name shows that this type would be useful for tabular data which is perfect for NoSQL data storage. This type of account has a stability feature and easy to use. Like other NoSQL data that is schema-less and accessible via REST API.
- Azure File storage: This type is used for legacy applications. Azure file storage provides file sharing in the cloud by using SMB protocol which supports both SMB3.0 and SMB2.1.
- Azure Queues storage: This kind of service is used to exchange message between components in the cloud and local data center. By using this account, it would be possible to keep large number of messages that should be shared between independent components of applications and communicated asynchronously via HTTP or HTTPS. (58)

3.29 Azure storage account

Azure storage account contains all azure storage data objects such as Blobs, files, queue, tables and disks. A specific namespace is provided by the Azure account that would be accessible from anywhere over HTTP or HTTPS. We have several account storages types that each of them has a specific feature and a pricing model. (57)

- General-purpose v2 accounts: This is one of the basic storage account types for blobs, files, queues and tables. It would be recommended for the most Azure storage scenarios. (57)
- General-purpose v1 accounts: That is an older version of General-purpose v2. (57)

- Block Blob Storage accounts: This type of account has high performance and would be useful for scenarios that need high transaction rates, small objects or low storage latency. (57)
- File storage accounts: This account would be useful for projects that need enterprise or high-performance scale applications. (57)
- Blob storage account: This is legacy blob-only storage account. (57)

3.29.1 General purpose storage account:

This type of account is very universal. It is included any type of available service such as blobs, files, queues and tables. During creation of this type of account, two options are available. Standard option which holds queues, tables, blobs and files. Another option is premium which is capable of storing azure virtual machine. (Modi, Damaschke, Klaffenbach, & Michalski, 2018)

3.29.2 Blob storage account:

Storage account provides access to Azure storage services such as virtual machine disks tables, queues, files, blobs and Azure combined in a single account. Two level of performances are available; (Modi, Damaschke, Klaffenbach, & Michalski, 2018)

- Standard storage performance tier: The standard performance storage level allows the customer to file tables, queues, files, blobs and Azure virtual machine disks.
- Premium storage performance tier: It currently supports virtual machine discs from Azure exclusively.

In order to keep unstructured data as objects, a Blob storage account is available in Azure storage. Blob storage shares features with existing storage accounts for general purpose. Microsoft Azure recommends that Blob storage accounts be used for applications that require blob storage to be completely blocked or appended. Blob storage accounts show the access level attribute that can be specified during the account creation process. Two types of access level can be defined based on data access pattern; (Modi, Damaschke, Klaffenbach, & Michalski, 2018)

- Hot access level: This rank shows that items are often collected in the storage account. It helps data to be processed at a lower cost of access. (Modi, Damaschke, Klaffenbach, & Michalski, 2018)
- Cool access: This rate implies less regular access to the items in the storage account. It has a low cost as well. (Modi, Damaschke, Klaffenbach, & Michalski, 2018)

- Archive access: This access level would be available only for individual block blobs. This kind of tier would be optimized for data that is optimized for several hours' latency and could be remain in archive for at least 180 days. This kind of access is the most effective option but accessing data is more expensive than accessing in hot or cold level. (59)

3.30 Replication:

There are several options for replication data in Azure network:

- LRS: Locally redundant storage is very simple and low-cost replication method that synchronized all data three times within the primary region.
- ZRS: Zone-redundant storage that replicates scenarios needs high availability. In this method data would be synchronized across three azure zones in the primary region.
- GRS: Geo-redundant storage which replicates data three times in the primary region then synchronized in the second region. Overly, this type of replication protects data against regional outage.
- GZRS: Geo-Zone-redundant storage would be useful for scenarios which need both high availability and durability. Data is synchronized three times in the primary region then replicated asynchronously to the second region. (59)

3.31 Azure storage encryption

When data is put on the cloud of Azure, automatically it would be encrypted by using 256-bit AES encryption methods which is similar to BitLocker of Windows. This feature is enabled by default for all type of storage accounts, resource managers, and classic storage accounts. Encryption cannot be disabled and there is no need to do any extra configuration. Encryption would be done without considering the performance or deployment model. Also, all storage replication models support encryption and all copies of data would be encrypted. Moreover, encryption would not have side effects on performance and users would not be charged. (59)

3.32 Access Key

In order to authenticate internal or external applications that interact with Azure storage, an access key is used. Two 512-bit access keys are generated while the storage account is created. These two keys are very important for security account and should be kept in a safe place. Due to high availability, there are two access keys in each storage account. (Modi, Damaschke, Klaffenbach, & Michalski, 2018)

Also, Microsoft provides a way to manage the keys and regenerate them regularly in order to protect the data by using Azure Key Vault. Indeed, the storage account access key is like root password, by regenerating them regularly, we are avoiding distributing access keys to other users, hard coding them or saving them anywhere in plain text that would be accessible to others. Also, Microsoft recommends using AD for authorizing requests from Blob and Queue storage instead of using shared key. (60)

3.33 Azure Virtual machine:

If there is a need to run a specific service that Azure doesn't have any offer for that, we could implement virtual machine in our environment but before any set up, it is necessary to know about different types of VMs in Azure. Azure has lots of offers for VMs because virtual machines are always the basements for every service offered out of Azure or other Microsoft cloud survives. Each type of virtual machine has different kinds of workloads. In the following, all type of virtual machines has been listed;

- For testing developing purpose we could use Basic A-series.
- For different workloads, we could use Standard A-series.
- For high performance computing, computing intense A-series would be a suitable option.
- D and DS-series would be useful when we have enterprise applications or applications that need high demand for compute power and temporary disk performance.
- F and FS-series that has been optimized for network applications such as web server or real time communication.
- G and GS-series would be a good option for high compute demand such as databases, SharePoint or big data calculations.
- H-series would be useful for high performance computing.
- N-series which includes an NVIDIA M60 and K80 for high GPU computing.
- Ls-series that has been designed for low latency storage demands.

For the virtual machine in which there is S in their name, those types represent the storage. These kinds of virtual machines use SSD and Azure premium storage as the primary storage for data and OS. (Modi, Damaschke, Klaffenbach, & Michalski, 2018)

3.33.1 A-series virtual machines

These series of virtual machine are categorized into two groups; standard and basic.

This type of virtual machine is very common in Microsoft deployments and there is a verity of hardware and CPU for different performance. At the following, some major differences between standard and basic have been illustrated; (Modi, Damaschke, Klaffenbach, & Michalski, 2018)

- Availability: The basic VMs are only available in small A0-A4 instances for testing purposes while standard versions are available on all size instances and regions.
- Disk IOPS: Data disk IOPS for basic type is up to 300 while for the standard type is up to 500.
- Price: The price for basic VMs is up to 27% less expensive than standard.
- Feature cut: Basic VMs do not have any load balancing or scaling options. In order to add these kinds of options, we need to add those in the availability set and implement our load balance mechanism.
- CPU: Standard level has better CPU performance comparing to basic level.
- Usage: The purpose of basic is testing and development while standard version is useful for a production. (Modi, Damaschke, Klaffenbach, & Michalski, 2018)

A standard version is a good option for common usages such as AD, federation services, or other basic network and application services.

3.33.2 D-series and DS-series virtual machine

D-series virtual machine has been designed for applications that need high compute and temporary disks. Their processors are higher comparing to A-series, and they use SSD as hard disk drive. They are suitable for some modern operating systems like Windows Server 2012R2 and they would be able to handle all workloads from file servers, databases, applications, and web servers.

3.33.3 F-series and FS-series Virtual machine

These VM series have 2.4 GHz Intel Xeon® E5-2673 v3 (Haswell) processor that could have clock speed as high as 3.1GH. These VMs are very suitable for workloads which needs fast CPU but not much memory or Hard disk capacity such as gaming, analytics, batch processing and so on.

3.33.4 G-series and GS-series Virtual machines

This type of virtual machine offers up to 32 vCPUs using the latest Intel® Xeon® processor E5 V3 family, 448 GB of memory, and 6.59 TB local SSD drives which would be meet the requirement for very large scale-up enterprise applications such as large relational database servers (SQL Server, MySQL, and so on) and large NoSQL databases (MongoDB, Cloudera,

Cassandra, and so on). This type is the highest virtual machine which offered by Azure. In the following, different type of virtual machine for these two series illustrated;

Size	CPU	Memory	HDD GB	Max disks	Max disk IOPS	Max NICs/Network bandwidth
G1	2	28	384	4	4 x 500	1/high
G2	4	56	768	8	8 x 500	2/high
G3	8	112	1,536	16	16 x 500	4/very high
G4	16	224	3,072	32	32 x 500	8/extremely high
G5	32	448	6,144	64	64 x 500	8/extremely high

Figure 3. 31

3.33.5 H-series Virtual machines:

This type of virtual machine with 8 and 16 cores on the Intel Haswell E5-2667 V3 processor technology with DDR4 memory and local SSD based storage would be suitable for high-end computational needs such as molecular modeling, and computational fluid dynamics such as wave calculations. Also, this type offers diverse options for low latency RDMA networking using FDR InfiniBand and different memory configurations to support memory-intensive computational requirements. In the following table, different models of this type have been listed;

Size	CPU	Memory	HDD GB	Max disks	Max disk IOPS	Max NICs/Network bandwidth
H8	8	56	1000	16	16 x 500	2/high
H16	16	112	2000	32	32 x 500	4/very high
H8m	8	112	1000	16	16 x 500	2/high
H16m	16	224	2000	32	32 x 500	4/very high
H16r	16	112	2000	32	32 x 500	4/very high
H16mr	16	224	2000	32	32 x 500	4/very high

Figure 3. 32

3.33.6 NV-series and NC-series virtual machines:

These two series have been equipped with NVIDIA's GPU cards which would meet the requirements for remote visualization, streaming, gaming, encoding and VDI scenarios such as with Citrix utilizing frameworks (NV-series). While NC- series would be used for compute-intensive and network-intensive applications and algorithms, including CUDA and OpenCL based applications and simulations. At the two following tables, different models of these two types have been listed;

Size	CPU	Memory	HDD GB	GPU
NV6	6	56	380	1 x NVIDIA M60
NV12	12	112	680	2 x NVIDIA M60
NV24	24	224	1440	4 x NVIDIA M60

Figure 3. 33

Size	CPU	Memory	HDD GB	GPU
NC6	6	56	380	1 x NVIDIA K80
NC12	12	112	680	2 x NVIDIA K80
NC24	24	224	1440	4 x NVIDIA K80

Figure 3. 34

3.33.7 LS-series Virtual machines:

This type of virtual machine offers low latency local storage with high demand for IOPS. It would be a good option for applications like NoSQL databases or virtualized Windows Server Storage Spaces Direct Servers.

3.34 Availability set

Availability set is used to separate VMs resources in this way if one set faces hardware or software issues, only VMs inside that set would have problems and other sets stays operational. Actually, availability sets are essential for building reliable cloud solutions. There are several issues which may have an impact on the availability of Azure virtual machines such as;

- Unplanned hardware maintenance event: If hardware fails, Azure fires an unplanned repair event for the hardware. In this kind of situation, live migration is used in order to move VM, network connections, memory and storage to different virtual machines without any disconnections for clients.
- Unexpected Downtime: When this event occurs, the virtual machine is down because Azure needs to heal VM inside the same data center.
- Planned hardware maintenance event: This type of event occurs periodically in order to update the platform. Most of them do not have any side effects on the uptime of VMs.

In order to provide redundancy for these types of events, we could group two or more VMs inside an availability set. In this case, during an event or failure, only a subnet of VMs is impacted and the remain cloud structure would operate normally. (Zaal, 2018)

3.35 Understanding cloud services architecture

The cloud services comprise of two elements;

- Cloud service package: This is a Zip file with ". cspkg" file extension which includes service definition file (.csdef) and code assets for services and required binary-base dependencies.
- Service configuration file: This file has ".cscfg" extension.

As it could be seen in the following diagram, service configuration file is the outside of the cloud service package, so every change in the configuration file could be made without interrupting running services while any changes to the services require a redeployment of the cloud service package.

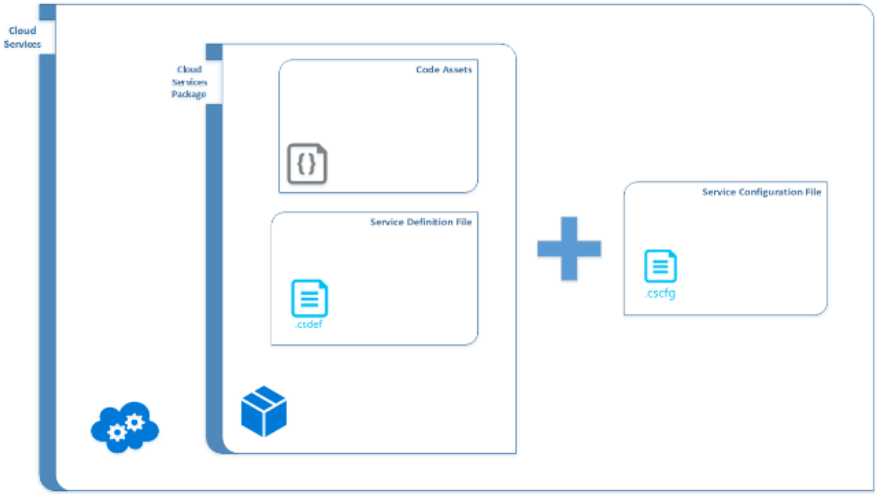

Figure 3. 35

3.36 Roles:

Another element in the cloud architecture is roles which are created by services definition files and code assets. There are two kinds of roles:

- WebRoles: It is referred to as cloud service instances that are running on VM Windows with installed IIS. Actually, WebRoels are hosting of web apps.
- WorkerRoles: It is referred to as cloud services that are running on VM without any installed IIS. This type of roles is responsible for internal processing of business logic and for communicating on Azure platforms.

Each cloud service has at least one WebRole or WorkRole otherwise the service is not reachable from outside.

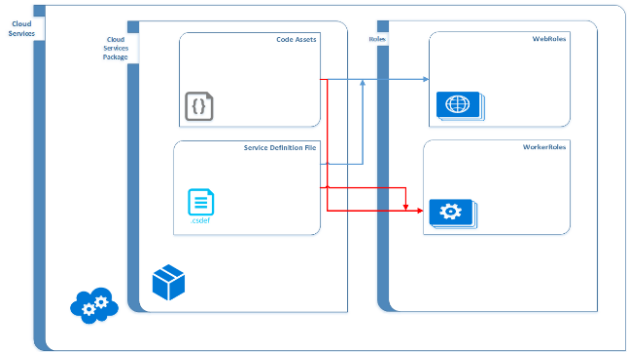

Figure 3. 36

3.37 The service endpoint:

The last element in a cloud service is service endpoint. This service is provided by a WebRole and it is the public interface to outside.

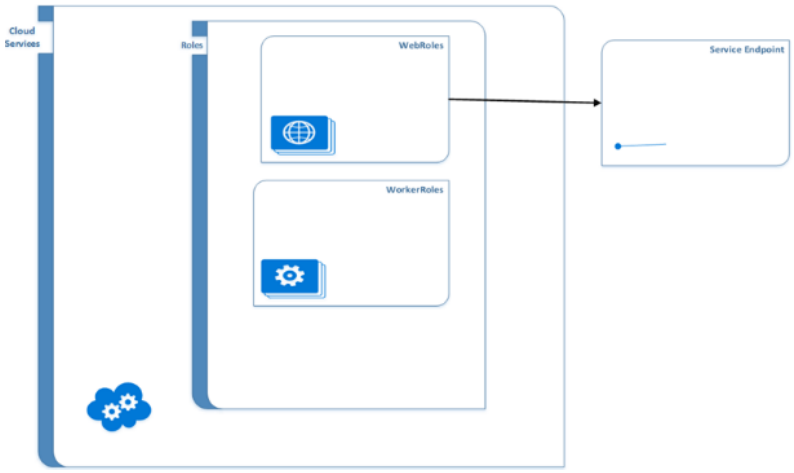

Figure 3. 37

3.38 Service Definition File

This is an XML file based on the Azure service definition scheme and it describes the components of the cloud services. There are five elements in this file;

- Root element: This is top level element with the mandatory attribute name and three optional attributes, topologyChangeDiscovery, schemaVersion, and upgradeDomainCount.
- LoadBalancerProbes: Defined in the Azure load balancer probe schema, this is an optional element.
- WebRole: Defined in the Azure WebRole scheme, this is a mandatory element.

- WorkerRole: Defined in the Azure WorkerRole scheme, this is a mandatory element.
- NetworkTrafficRules: Defined in the Azure network traffic rules scheme, this is an optional element.

3.39 LoadBalancerProbes

First, we need to clarify what Loadbalancerprobes is. Azure Load balancer which is identified by abbreviation ALB is responsible for routing incoming traffic to the role instance. But ALB should send a query to the respective endpoints and check that the URL returns a HTTP 200 OK code. This process is called the Load balancer probe. A lodaBalancerprobe is not an element by itself and usually it comes with combination WebRole or WorkerRole. In the following, a template of LoadBalancerprobe is shown;

```
<ServiceDefinition ...>
   <LoadBalancerProbes>
      <LoadBalancerProbe name="<load-balancer-probe-name>"
       protocol="[http | tcp]"
       path="<uri-for-checking-health-status-of-vm>"
       port="<port-number>" intervalInSeconds="<interval-in-seconds>"
       timeoutInSeconds="<timeout-in-seconds>"/>
   </LoadBalancerProbes>
</ServiceDefinition>
```

Figure 3. 38

The attributes name and protocol are mandatory. Regarding attribute path, it would be essential when value attribute protocol is set to HTTP. Other attributes are optional.

3.40 WebRole:

By using a Webrole element, we determine a web application. In the following, a template of WebRole is shown;

```
<ServiceDefinition ...>
  <WebRole name="<web-role-name>" vmsize="<web-role-size>"
   enableNativeCodeExecution="[true | false]">
    <Certificates>
      <Certificate name="<certificate-name>" storeLocation="
       <certificate-store>" storeName="<store-name>" />
    </Certificates>
    <ConfigurationSettings>
```

```
  <Setting name="<setting-name>" />
 </ConfigurationSettings>
 <Imports>
  <Import moduleName="<import-module>"/>
 </Imports>
 <Endpoints>
  <InputEndpoint certificate="<certificate-name>"
   ignoreRoleInstanceStatus="[true | false]"
   name="<input-endpoint-name>" protocol="[http| https| tcp| udp]"
   localPort="<port-number>" port="<port-number>"
   loadBalancerProbe="<load-balancer-probe-name>" />
  <InternalEndpoint name="<internal-endpoint-name>"
   protocol="[http | tcp | udp | any]" port="<port-number>">
    <FixedPort port="<port-number>"/>
    <FixedPortRange min="<minium-port-number>"
     max="<maximum-port-number>"/>
  </InternalEndpoint>
  <InstanceInputEndpoint name="<instance-input-endpoint-name>"
   localPort="<port-number>" protocol="[udp | tcp]">
    <AllocatePublicPortFrom>
      <FixedPortRange min="<minium-port-number>"
       max="<maximum-port-number>"/>
    </AllocatePublicPortFrom>
  </InstanceInputEndpoint>
 </Endpoints>
 <LocalResources>
  <LocalStorage name="<local-store-name>"
   cleanOnRoleRecycle="[true | false]"
   sizeInMB="<size-in-megabytes>" />
 </LocalResources>
 <LocalStorage name="<local-store-name>"
```

```
 cleanOnRoleRecycle="[true | false]"
 sizeInMB="<size-in-megabytes>" />
<Runtime executionContext="[limited | elevated]">
 <Environment>
   <Variable name="<variable-name>" value="<variable-value>">
     <RoleInstanceValue
      xpath="<xpath-to-role-environment-settings>"/>
   </Variable>
 </Environment>
 <EntryPoint>
   <NetFxEntryPoint
    assemblyName="<name-of-assembly-containing-entrypoint>"
    targetFrameworkVersion="<.net-framework-version>"/>
 </EntryPoint>
</Runtime>
<Sites>
 <Site name="<web-site-name>">
   <VirtualApplication name="<application-name>"
    physicalDirectory="<directory-path>"/>
   <VirtualDirectory name="<directory-path>"
    physicalDirectory="<directory-path>"/>
   <Bindings>
     <Binding name="<binding-name>"
      endpointName="<endpoint-name-bound-to>"
      hostHeader="<url-of-the-site>"/>
   </Bindings>
 </Site>
</Sites>
<Startup priority="<for-internal-use-only>">
 <Task commandLine="<command-to=execute>"
  executionContext="[limited | elevated]"
```

```
      taskType="[simple | foreground | background]">
       <Environment>
        <Variable name="<variable-name>" value="<variable-value>">
          <RoleInstanceValue
           xpath="<xpath-to-role-environment-settings>"/>
        </Variable>
       </Environment>
     </Task>
   </Startup>
   <Contents>
    <Content destination="<destination-folder-name>" >
     <SourceDirectory path="<local-source-directory>" />
    </Content>
   </Contents>
  </WebRole>
</ServiceDefinition>
```

In the following, you could see the key elements of WebRole;

- Sites: It determines a collection for websites or web applications that are hosted in IIS. If any element is not defined for the site, it means that we have only one website or web application.
- Site (Child of sites element): Determines a definition for a website or web application which is hosted in IIS.
- Endpoints: It describes a definition for input, internal and instance input endpoints for a role.
- InputEndpoints (child of Endpoints element): It describes an external endpoint that is responsible for contacting cloud services. We could define HTTP, HTTPS, UDP and TCP as an endpoint here.
- InstanceInputEndpoint (Child of Endpoint element): It used for describing instance input end points which is used for port forwarding in Azure Load balancer.
- Fixedport: It determines a port for an internal endpoint which provides connections to Azure load Balancer.

- FixedPortRange: It used to specify a range port for internal input endpoints.
- ConfigurationSettings: It describes settings for features.
- Certificates: It contains certificates that are needed for the role.
- Imports: It describes definitions for imported modules.
- VirtualApplication: When a virtual application is created in IIS, the application's path becomes part of site's URL.
- VirtualDirectory: It used to map a physical directory in IIS.
- Startup: It describes some operations to perform before a role is started.

3.41 WorkerRole:

By using workerRole, we could able to determine tasks for background processing inside from a WebRole process. In the following, a template of WorkerRole has been shown;

```
<ServiceDefinition ...>
 <WorkerRole name="<worker-role-name>" vmsize="<worker-role-size>"
  enableNativeCodeExecution="[true | false]">
   <Certificates>
     <Certificate name="<certificate-name>"
      storeLocation="[CurrentUser | LocalMachine]
      storeName="[My|Root|CA|Trust|Disallow|TrustedPeople|
      TrustedPublisher|AuthRoot|AddressBook|<custom-store>]" />
   </Certificates>
   <ConfigurationSettings>
     <Setting name="<setting-name>" />
   </ConfigurationSettings>
   <Endpoints>
     <InputEndpoint name="<input-endpoint-name>"
      protocol="[http | https | tcp | udp]"
      localPort="<local-port-number>" port="<port-number>"
      certificate="<certificate-name>"
      loadBalancerProbe="<load-balancer-probe-name>" />
     <InternalEndpoint name="<internal-endpoint-name"
      protocol="[http | tcp | udp | any]" port="<port-number>">
       <FixedPort port="<port-number>"/>
       <FixedPortRange min="<minium-port-number>"
        max="<maximum-port-number>"/>
     </InternalEndpoint>
     <InstanceInputEndpoint name="<instance-input-endpoint-name>"
      localPort="<port-number>" protocol="[udp | tcp]">
       <AllocatePublicPortFrom>
         <FixedPortRange min="<minium-port-number>"
          max="<maximum-port-number>"/>
       </AllocatePublicPortFrom>
     </InstanceInputEndpoint>
   </Endpoints>
   <Imports>
     <Import moduleName=
      "[RemoteAccess |RemoteForwarder | Diagnostics]"/>
   </Imports>
```

```
    <LocalResources>
      <LocalStorage name="<local-store-name>"
       cleanOnRoleRecycle="[true | false]"
       sizeInMB="<size-in-megabytes>" />
    </LocalResources>
    <LocalStorage name="<local-store-name>"
     cleanOnRoleRecycle="[true | false]"
     sizeInMB="<size-in-megabytes>" />
    <Runtime executionContext="[limited | elevated]">
      <Environment>
        <Variable name="<variable-name>" value="<variable-value>">
          <RoleInstanceValue
           xpath="<xpath-to-role-environment-settings>"/>
        </Variable>
      </Environment>
      <EntryPoint>
        <NetFxEntryPoint
         assemblyName="<name-of-assembly-containing-entrypoint>"
         targetFrameworkVersion="<.net-framework-version>"/>
        <ProgramEntryPoint
         commandLine="<application>"
         setReadyOnProcessStart="[true |false]" "/>
      </EntryPoint>
    </Runtime>
    <Startup priority="<for-internal-use-only>">
      <Task commandLine="" executionContext="[limited | elevated]"
       taskType="[simple | foreground | background]">
        <Environment>
        <Variable name="<variable-name>" value="<variable-value>">
          <RoleInstanceValue
           xpath="<xpath-to-role-environment-settings>"/>
         </Variable>
        </Environment>
      </Task>
    </Startup>
    <Contents>
      <Content destination="<destination-folder-name>" >
        <SourceDirectory path="<local-source-directory>" />
      </Content>
    </Contents>
  </WorkerRole>
</ServiceDefinition>
```

All the elements in this Role would be the same as WebRole.

3.42 NetworkTraffcRules:

By using this role, we could be able to specify how a role communicates with other roles.

Actually, it could limit which role has access the internal endpoints of the specific role.

Also, it is not a standalone and usually, it combines with a WebRole or WorkerRole. A template of this role has been shown in the following;

```
<ServiceDefinition ...>
   <NetworkTrafficRules>
      <OnlyAllowTrafficTo >
         <Destinations>
            <RoleEndpoint endpointName="<name-of-the-endpoint>"
             roleName="<name-of-the-role-containing-the-endpoint>"/>
         </Destinations>
         <AllowAllTraffic/>
         <WhenSource matches="[AnyRule]">
            <FromRole
             roleName="<name-of-the-role-to-allow-traffic-from>"/>
         </WhenSource>
      </OnlyAllowTrafficTo>
   </NetworkTrafficRules>
</ServiceDefinition>
```

Figure 3. 39

In the following, these elements have been illustrated;

- OnlyAllowTraffic: It contains a collection of endpoints and the roles that communicate with them.
- Destinations: It specifies a collection of RoleEndpoint.
- RoleEndpoint: It determines an endpoint on a role.
- AllowAllTraffic: It determines a rule that allows all roles to communicate with the endpoints which is defined in the Destination node.
- WhenSource: Determines a collection of roles that can communicate with the endpoints.
- FromRole: It contains the roles that can communicate with the endpoints.

3.43 Service configuration File:

This is an XML file based on Azure Service configuration Scheme and determines how the service is configured. In the following, a template of this file is shown;

<ServiceConfiguration serviceName="<service-name>" osFamily="<osfamily-number>" osVersion="<os-version>" schemaVersion="<schema-version>">

<Role>

</Role>

<NetworkConfiguration>

</NetworkConfiguration>

</ServiceConfiguration>

There are three elements in this file;

- Root element that is “<serviceConfiguration> “, service name is a mandatory attribute for this element. Also this root element has three optional attributes, osFamily,osVersion, and schemaVersion.
- Role: This element determines the number of role instances, the value of configuration settings and certificates.
- NetworkConfiguration: This element determines a virtual network and DNS value.

In the following, a template of Role has been shown;

```
<ServiceConfiguration>
  <Role name="<role-name>" vmName="<vm-name>">
    <Instances count="<number-of-instances>"/>
    <ConfigurationSettings>
      <Setting name="<setting-name>" value="<setting-value>" />
    </ConfigurationSettings>
    <Certificates>
      <Certificate name="<certificate-name>"
       thumbprint="<certificate-thumbprint>"
       thumbprintAlgorithm="<algorithm>"/>
    </Certificates>
  </Role>
</ServiceConfiguration>
```

Figure 3. 40

- Name: Determines the name of the role.
- VmName: It is used to determine a DNS for VM.
- Instances: The number of instances is determined in this section.
- Setting: It specifies a setting name and value.
- Certificate: it specifies certificate service.

In the following, a template of NetworkConfigurtion has been shown;

```
<ServiceConfiguration>
  <NetworkConfiguration>
    <AccessControls>
      <AccessControl name="aclName1">
        <Rule order="<rule-order>"
         action="<rule-action>" remoteSubnet="<subnet-address>"
```

```
        description="rule-description"/>
      </AccessControl>
    </AccessControls>
    <EndpointAcls>
      <EndpointAcl role="<role-name>" endpoint="<endpoint-name>"
       accessControl="<acl-name>"/>
    </EndpointAcls>
    <Dns>
      <DnsServers>
        <DnsServer name="<server-name>" IPAddress="<server-address>" />
      </DnsServers>
    </Dns>
    <VirtualNetworkSite name="<site-name>"/>
    <AddressAssignments>
      <InstanceAddress roleName="<role-name>">
        <Subnets>
          <Subnet name="<subnet-name>"/>
        </Subnets>
      </InstanceAddress>
      <ReservedIPs>
        <ReservedIP name="<reserved-ip-name>"/>
      </ReservedIPs>
    </AddressAssignments>
  </NetworkConfiguration>
</ServiceConfiguration>
```

- AccessControl: It determines the rules for accessing to the endpoints.
- Rule: Specifies the action that should be taken for a specific subnet range of IP addresses.
- EndpointAc1: It determines the relation between access control and endpoint.
- DnsServer: It specifies DNS setting
- VirtualNetworkSite: Name of virtual network.
- InstanceAddress: It determines the association of a role to a subnet.

- Subnet: Specifies a subnet.
- ReservedIP: Determines the reserved IP address that is associated to deployment.

4. AWS

AWS cloud offers a cutting-edge platform for architecting, building and deploying web-scale cloud application through user friendly interface. Due to variety of features that is available within AWS, overall cost is very reasonable and the speed of the development process for both large enterprise and start-up is increased.

4.2 AWS Console

In order to login to console, first we need to browse the address; https://aws.amazon.com, then we could be able to login by using Email address which used for creating account. It is root account. Or we would be able to login by using IAM user for creating accounts. IAM users create a console to give login and management permission to other users. Through Dashboard or the service menu, we could find the desire service. Also, those services which are used frequently, could be added to tab bar.

All billing information related to this month, pervious months and projection of next month can be found under my name menu and selecting “My billing dashboard”. (Curator, 2017)

4.3 AWS Component

AWS offers a variety of infrastructural services including compute, storage, databases, networking, mobile, developer tools, management tools, IOT and security. These services would be helpful for an organization to move fast, reduce IT costs and scale. (67)

4.3.1 Elastic Compute Cloud (EC2)

EC2 is a web service that provides compute capacity. With using EC2, we could be able to combine an operating system, application software and desire settings into an AMI. The number of EC2 could be increased or decreased based on the requirements. These instances could be launched in one or more geographical locations or regions and Availability Zones (AZs). Each region comprises of several AZs at distinct locations which are connected to each other by low latency networks in the same region. (Shah & Sarkar, Learning AWS, 2015)

4.3.1.2 T2

T2 instances are a low-cost, the purpose of this instance is for the general purpose since it uses share CPU with other instances instead of dedicating. They operate at low CPU performance, and when it idle, it accumulates credits. (4, n.d.) (Gilchrist & Soni, 2018)

Model	vCPU	CPU Credits / hour	Mem (GiB)	Storage
t2.nano	1	3	0.5	EBS-only
t2.micro	1	6	1	EBS-only
t2.small	1	12	2	EBS-only
t2.medium	2	24	4	EBS-only
t2.large	2	36	8	EBS-only
t2.xlarge	4	54	16	EBS-only
t2.2xlarge	8	81	32	EBS-only

Figure 4. 1

4.3.1.3 M4

M4 instances use custom Intel Xeon E5-2676 v3 Haswell processors that are optimized specifically for EC2. M4 instances use EBS which provides 450Mbps to 4000Mbps. Also, it should be mentioned that this output is depends on network card which is using on these instances. M4 has enhanced networking feature that results in throughput up to four times of instances without enhanced networking. This instance type is very suitable for small-and mid-sized databases and many web applications. (Gilchrist & Soni, 2018) (2, n.d.)

Model	vCPU	Mem (GiB)	SSD storage (GB)	Dedicated EBS bandwidth (Mbps)
m4.large	2	8	EBS-only	450
m4.xlarge	4	16	EBS-only	750
m4.2xlarge	8	32	EBS-only	1000
m4.4xlarge	16	64	EBS-only	2000
m4.10xlarge	40	160	EBS-only	4000
m4.16xlarge	64	256	EBS-only	10000

Figure 4. 2

(Gilchrist & Soni, 2018)

4.3.1.4 M3:

It is older version of M and used IVY Bridge processor. It provides up to 160GB storage. (Curator, 2017)

Model	vCPU	Mem (GiB)	SSD Storage (GB)
m3.medium	1	3.75	1 x 4
m3.large	2	7.5	1 x 32
m3.xlarge	4	15	2 x 40
m3.2xlarge	8	30	2 x 80

Figure 4. 3

4.3.2 Compute Optimized

The purpose of Compute optimized instances is for applications that need high-performance processors such as: (7)

- Batch processing workloads.
- Media transcoding.
- High-performance web servers
- High-performance computing (HPC)
- Scientific modeling
- Dedicated gaming servers and ad serving engines.
- Machine learning inference and other compute-intensive applications. (7)

4.3.2.1 C4

It uses the most powerful Haswell processors. They are EBS optimized and support enhanced networking and clustering. They could not be found in any instance storage. They are great high-performance front end fleets, analytics, and batch processing. (Curator, 2017) (Gilchrist & Soni, 2018)

Model	vCPU	Mem (GiB)	Storage	Dedicated EBS bandwidth (Mbps)
c4.large	2	3.75	EBS-only	500
c4.xlarge	4	7.5	EBS-only	750
c4.2xlarge	8	15	EBS-only	1000
c4.4xlarge	16	30	EBS-only	2000
c4.8xlarge	36	60	EBS-only	4000

Figure 4. 4

4.3.2.2 C5

The purpose of C5 instances is for machine learning, deep learning, batch processing, HPC. This compute instance provides enhanced networking (up to 25 Gbs network bandwidth). It uses EBS as storage. (Gilchrist & Soni, 2018)

Model	vCPU	Mem (GiB)	Instance storage (GiB)	Dedicated EBS bandwidth (Mbps)
c5.large	2	4	EBS-only	Up to 3,500
c5.xlarge	4	8	EBS-only	Up to 3,500
c5.2xlarge	8	16	EBS-only	Up to 3,500
c5.4xlarge	16	32	EBS-only	3500
c5.9xlarge	36	72	EBS-only	7000
c5.18xlarge	72	144	EBS-only	14000
c5d.large	2	4	1 x 50 NVMe SSD	Up to 3,500
c5d.xlarge	4	8	1 x 100 NVMe SSD	Up to 3,500
c5d.2xlarge	8	16	1 x 200 NVMe SSD	Up to 3,500
c5d.4xlarge	16	32	1 x 400 NVMe SSD	3500
c5d.9xlarge	36	72	1 x 900 NVMe SSD	7000
c5d.18xlarge	72	144	2 x 900 NVMe SSD	14000

Figure 4. 5

4.3.3 Memory-optimized instance

Memory optimized instances are suitable for applications that process large data sets in memory, such as; (8)

- High-performance, relational (MySQL) and NoSQL (MongoDB, Cassandra) databases.
- Distributed web scale cache stores that provide in-memory caching of key-value type data (Memcached and Redis).
- In-memory databases using optimized data storage formats and analytics for business intelligence (for example, SAP HANA).
- Applications performing real-time processing of big unstructured data (financial services, Hadoop/Spark clusters).
- High-performance computing (HPC) and Electronic Design Automation (EDA) applications. (8)

4.3.3.1 R5

This is an instance of memory-optimized instances that are suitable for memory intensive applications such as high-performance databases, distributed web scale in-memory caches, mid-size in-memory databases, real time big data analytics, and other enterprise applications. (9)

4.3.4 High Memory instance

High memory instances offer 6 TiB, 9 TiB, 12 TiB, 18 TiB, and 24 TiB of memory per instance. These instances are designed to run large in-memory databases, including production installations of SAP HANA. (8)

4.3.4.1 X1

X1 instances are perfect for in-memory database like SAP HANA and big data processing. Some of its features are as follows;

- Haswell processors
- SSD-base instance storage
- Lowest price per GB of RAM (Gilchrist & Soni, 2018)

Model	vCPU	Mem (GiB)	SSD storage (GB)	Dedicated EBS bandwidth (Mbps)
x1.16xlarge	64	976	1 x 1,920	7000
x1.32xlarge	128	1952	2 x 1,920	14000

Figure 4. 6

4.3.5 Accelerated computing

Accelerated computing instance families use hardware accelerators, or co-processors, to perform some functions, such as floating-point number calculations, graphics processing, or data pattern matching. (6)

4.3.5.1 G3

G3 instances use NVIDIA Tesla M60 GPUs which would be useful for graphics applications using DirectX or OpenGL. G3 instances also provide NVIDIA GRID Virtual Workstation features which support four monitors with resolutions up to 4096x2160, and NVIDIA GRID Virtual Applications. The purpose of G3 instances is for applications such as 3D visualizations, graphics-intensive remote workstations, 3D rendering, video encoding, virtual reality, and other server-side graphics workloads that requiring massively parallel processing power (6)

4.3.6 Storage Optimizes instances

The purpose of Storage optimized instances is for workloads that require high, sequential read and write access to very large data sets on local storage. They are designed in which to deliver tens of thousands of low-latencies, random I/O operations per second (IOPS) to applications. (10)

4.3.6.1 I3

These instances are well suited for the following applications:

- High frequency online transaction processing (OLTP) systems.
- Relational databases.
- NoSQL databases.

- Cache for in-memory databases (for example, Redis).
- Data warehousing applications.
- Distributed file systems (10)

Refer to the below I3 instance details

Model	vCPU	Mem (GiB)	Networking performance	Storage (TB)
i3.large	2	15.25	Up to 10 Gigabit	1 x 0.475 NVMe SSD
i3.xlarge	4	30.5	Up to 10 Gigabit	1 x 0.95 NVMeSSD
i3.2xlarge	8	61	Up to 10 Gigabit	1 x 1.9 NVMe SSD
i3.4xlarge	16	122	Up to 10 Gigabit	2 x 1.9 NVMe SSD
i3.8xlarge	32	244	10 Gigabit	4 x 1.9 NVMe SSD
i3.16xlarge	64	488	25 Gigabit	8 x 1.9 NVMe SSD
i3.metal	72*	512	25 Gigabit	8 x 1.9 NVMe SSD

Figure 4. 7

4.3.6.2 D2

D2 instances provide high disk throughput, high performance at launch time and enhanced networking, Also, they are well suited for the following applications:

- Massive parallel processing (MPP) data warehouse
- MapReduce and Hadoop distributed computing
- Log or data processing applications (10, n.d.) (Gilchrist & Soni, 2018)

Refer to the below D2 instance details.

Model	vCPU	Mem (GiB)	Storage (GB)
d2.xlarge	4	30.5	3 x 2000 HDD
d2.2xlarge	8	61	6 x 2000 HDD
d2.4xlarge	16	122	12 x 2000 HDD
d2.8xlarge	36	244	24 x 2000 HDD

Figure 4. 8

4.3.7 Storage

EC2 provides flexible, cost-effective, and easy-to-use data storage options for the instances. Each option has a unique combination of performance and durability. These storage options can be used independently or in combination. (3)

The following figure illustrates the relationship between these storage options and the instance.

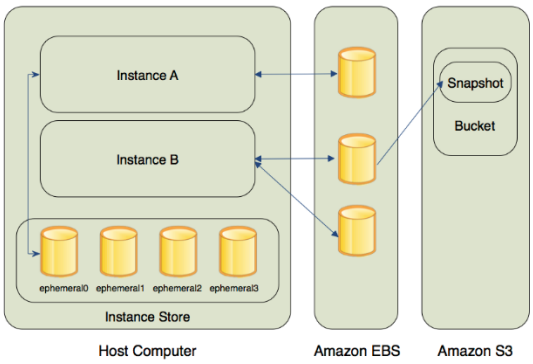

Figure 4. 9

4.3.7.1 S3

S3 is a reliable and inexpensive data storage infrastructure. The purpose of S3 is storing and retrieving any amount of data, at any time, from within Amazon EC2 or anywhere on the web. Amazon S3 is used to store backup copies of data and applications. Amazon EC2 uses Amazon S3 to store EBS snapshots and instance store-backed AMIs. (3)

4.3.7.2 EBS

An EBS (Elastic Block Store) volume behaves like a raw, unformatted, external block device that would be attached to an instance. Volume persists independently from the running life of an instance. (Patel, n.d.). After an EBS volume is attached to an instance, it is like any other physical hard drive. Also, it is possible to detach an EBS volume from one instance and attach it to another instance. To keep a backup copy of data, it is possible to create a snapshot of an EBS volume, which is stored in Amazon S3. (3)

4.3.7.3Instance Store

It is ephemeral storage that provides temporary block-level storage for an instance. So, data on this type of disk depends on the life of an associated instance. In which, if the instance is stopped or terminated, all data would be lost as well. (Patel)

4.3.8 EC2 Key Pair

Pair key (public and private key) would be used in Amazon EC2 in order to encrypt and decrypt data. Public key is used to encrypt data and private key is essential for decryption. The public and private keys are known as a key pair. Public key is needed to access the instances in a secure way while by using a private key as a password. When an instance is launched, the key pair should be identified. (16)

4.3.9 Security groups

A security group acts as a virtual firewall that controls the traffic for one or more instances. When an instance is launched, one or more security groups could be identified for them. Otherwise, the default security group would be applied. Rules can be added to each security group and the rules of each security group would be modified at any time. It should be considered that the new rules are automatically applied to all instances that are associated with the security group. (5)

4.3.10 AMI

An Amazon Machine Image (AMI) provides the information that is required to launch an instance. In order to launch an instance, one AMI should be specified. It is possible to launch multiple instances from a single AMI. An AMI includes the following information:

- One or more EBS snapshots
- Launch permissions that control which AWS accounts can use the AMI to launch instances.
- A block device mapping that identifies the volumes to attach to the instance when it's launched. (15)

The following diagram shows the whole lifecycle of AMI. After creating and registering an AMI, it can be used to launch new instances. An AMI can be copied within the same region or to different regions. (15)

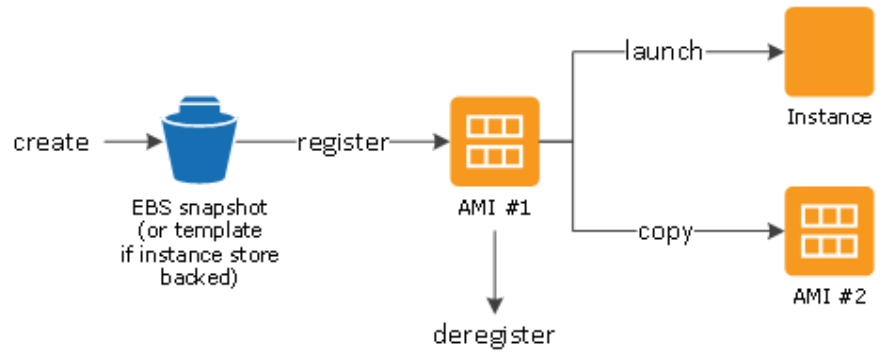

Figure 4. 10

4.4 Networking

AWS provides networking tools and resources which would be useful to connect to the cloud and isolate and control our applications in a secure way.

4.4.1 VPC

VPC or Amazon virtual private cloud specifies a virtual network that we can launch our EC2 resources such as instances into the subnets of a VPC. Actually, VPC is closely to the traditional network that we would operate in our own data center but with the benefits of using scalable infrastructure from AWS. Also, we can configure own VPC by selecting our desire range IP address, creating subnets, and configuring route tables, network gateways, and security settings. Moreover, instances in a PVC can be connected to the internet or to own data center. When an account is created in AWS, a default VPC is created. A default VPC is a VPC that is already configured and ready to use. It is possible to create a none-default VPC and configure it based on the requirement instead of using the default VPC.

4.4.2 Instance IP address

EC2 and VPC support both IPV4 and IPV6. IPV4 could not be disabled from EC2 and VPC. When a VPC is created, a range IP address should be assigned to that VPC but assigning IPV6 is optional. Generally, when an instance is created, a private IP address is allocated, and a DNS hostname is set as well. This DNS is sued to communicate between instances in the same VPC but cannot be used the outside of the VPC. We could be able to set secondary private IP address as well that unlike primary IP address, the secondary could be reassigned from one instance to another. A private IPv4 address, regardless of whether it is a primary or secondary address, remains associated with the network interface when the instance is stopped and restarted. It is released when the instance is terminated.

Each instance can have a public IP address which is used for external DNS hostname. An external DNS hostname is used to resolve a public IP address. The public IP address is mapped to the primary private IP address through network address translation (NAT). When an instance is created in a default VPC, a public IP address is assigned to it by default. A public IP address is assigned to an instance from Amazon's pool of public IPv4 addresses and is not associated with AWS account. When a public IP address is disassociated from that instance, it is released back into the public IPv4 address pool, and it cannot be used again. (17)

4.4.3 Elastic IP Addresses (IPv4)

An elastic IP is independent of the instance lifecycle. It is given to the account, not a specific instance. One of the benefits of this elastic IP address is that it could be given to an instance that already exists, even if it is already running. When instance is stopped, this IP address is not disassociated. So, it can be freely stopped and started EC2 instances with elastic IPs, and they will retain the same public address. By default, requesting up to five Elastic IPs per region is allowed. One Elastic IP is free per instance. (Gilchrist & Soni, 2018)

4.4.4 Elastic network interface

Each EC2 has one or more ENI (Elastic network interface). When an instance is created, a default ENI which is called eth0 is assigned to instance. This ENI cannot be detached but more ENI would be added to instances and those secondary ENI would be moved from one instance to another. So, if we want to replace an impaired instance with a new one with the same IP address, we would be able to move secondary ENI between these two instances. In this case, there is no need to update DNS or application configuration file.

All security groups are associated with the ENI and are not directly associated with the instance. (Gilchrist & Soni, 2018)

4.4.5 Route table

A rout table includes a set of rules which determines where network traffic from a subnet or gateway is directed. In the following, you would find some key concepts regarding route table;

- Main route table: This rout table created automatically for each VPC. It controls the routing for all subnets that are not explicitly associated with any other route table.
- Custom route table: A route table that we create for our VPC.
- Route table association: The association between a route table and a subnet, internet gateway, or virtual private gateway.
- Subnet route table: A route table that it is associated with a subnet.
- Gateway route table: A route table that it is associated with an internet gateway or virtual private gateway.
- Local gateway route table: A route table that it is associated with an Outposts local. gateway.
- Destination: The destination CIDR.
- Target: The target through which to send the destination traffic.
- Local route: A default route for communication within the VPC.

Each VPC has an implicit router that uses route tables to control where network traffic is directed. Each subnet in a VPC should be linked to a route table. It is possible to link a subnet with a particular rout table. Otherwise, the subnet is implicitly associated with the main route table. A subnet can only be associated with one route table at a time, but multiple subnets can be associated with the same subnet route table. A route table could be linked with an internet gateway or a virtual private gateway. This feature provides us a possibility to specify routing rules for inbound traffic that enters the VPC through the gateway. (18)

4.4.5.1 Main Route table

When we create a VPC, a main rout table is created by default. This main route table cannot be deleted but it can be replaced with a custom subnet rout table. This main route table controls all the routing for all subnets which are not associated with any other route table.

On the “Route Tables” page in the Amazon VPC console, the main route table for a VPC is shown by looking for “Yes” in the “Main” column. By default, when we create a non-default VPC, the main route table includes only a local route. When we use the VPC wizard in the console to create a none-default VPC with a NAT gateway or virtual private gateway, the wizard automatically adds routes to the main route table for those gateways. Adding, removing, and modifying routes in the main route table is possible. (18)

4.4.5.2 Custom Route tables

By default, a custom route table is empty, but we would be able to add routes to this custom table. By using VPC wizard in the console in order to create a VPC with an internet gateway, the wizard creates a custom route table and adds a route to the internet gateway. One of the ways to protect our PVC is that leave the main table, in this way we would make sure how traffic is controlled on each subnet. We can add a route to custom route table, but we cannot delete a custom routable while it is associated with one subnet. (18)

4.4.6 Implicit and Explicit Subnet Association

In the following, there is a diagram which clearly shows the routing for a VPC. The main route table including a route which is used for a virtual private gateway. In order to access to the internet, a custom route table has been associated with a public subnet.

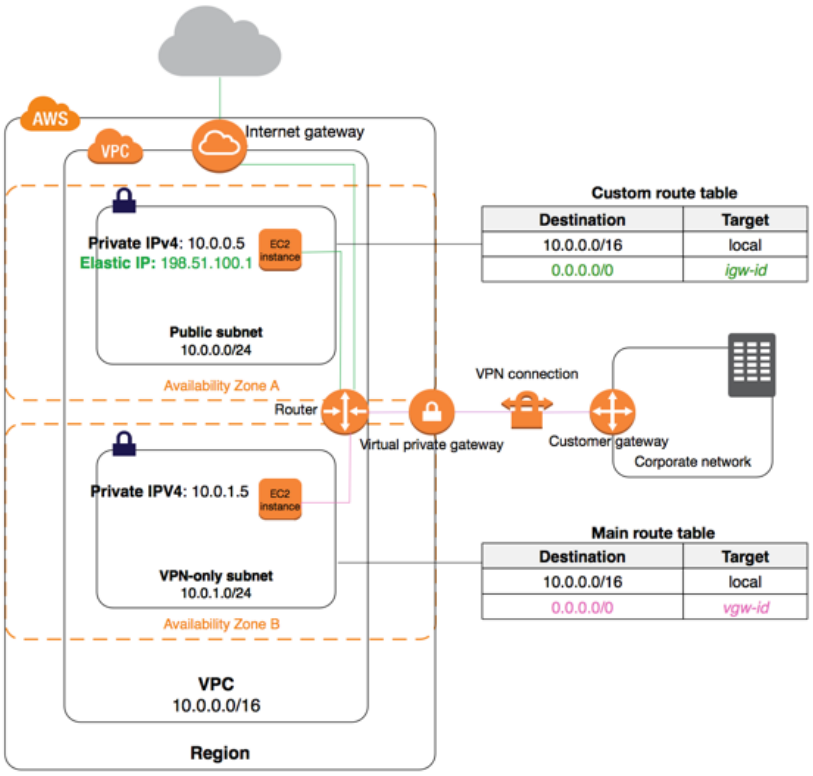

Figure 4. 11

If we create a new subnet in this VPC, it automatically associates with the main route table, which routes traffic to the virtual private gateway. (18)

4.4.7 Replacing the Main Route Table

One of the ways to replace main route table is that creating a custom route table and making sure everything is working well then replace the custom route table with the main route table.

The following diagram shows a VPC with two subnets that are linked to the main route table (Route Table A), and a custom route table (Route Table B) that isnot associated with any subnets. (18)

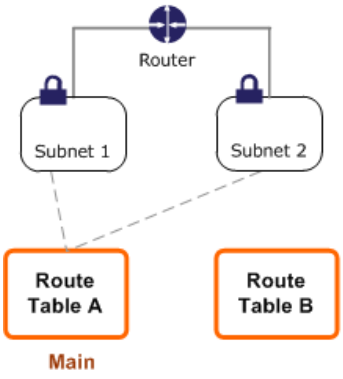

Figure 4. 12

Creating an explicit association between Subnet 2 and Route Table B.

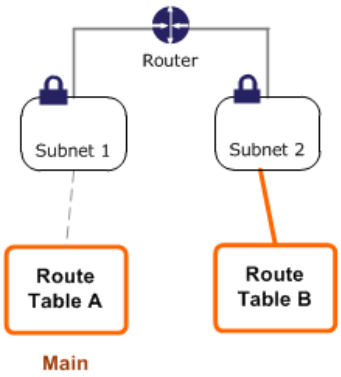

Figure 4. 13

After Route Table B is tested, we can make it as the main route table. Just we need to consider that Subnet 2 has been associated with Route table B and subnet 1 has been associated implicitly to Route table B. (18)

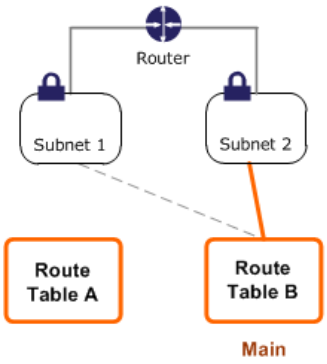

Figure 4. 14

We can disassociate subnet 2 and Route Table B and still there is an implicit association between Subnet 2 and Route Table B. After that we would delete Route Table A. (18)

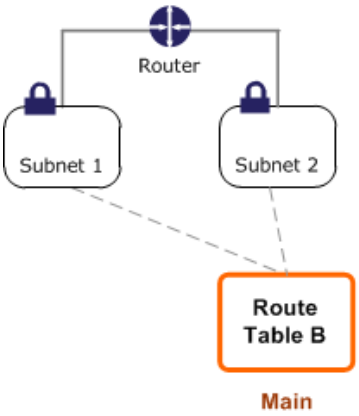

Figure 4. 15

4.4.8 Gateway Route Tables

Each route table which has been associated with an internet gateway or a virtual private gateway, is called a gateway route table. One of the main purposes of creating a gateway route table is controlling over the traffic entering the VPC. We can create a route in gateway which redirect to security devices in the network. (18)

4.4.9 NAT

By using NAT device, instances within private subnet would have access to the internet or other AWS services. Actually, NAT devices would forward traffic which comes from a private subnet to the internet by replacing the source IP address with own IP address then send the responds to the private from the internet as well. Just it should be mentioned that NAT devices do not support for IPv6 traffic. (19)

In AWS we have two kind of NAT device;

- NAT Gateway
- NAT Instance

4.4.9.1 NAT Gateway

In order to create a NAT gateway, we need the public subnet and one Elastic IP address which is associated with the NAT gateway. The Elastic IP address cannot be changed once it is associated. After creating a NAT gateway, we need to update the route table in order instances inside a private zone could have connection to the internet. Each NAT is for one zone and usually it would be with a redundancy.

In the following, there is a diagram which clearly shows what happened, when we use NAT gateway. Actually, the main route table sends the traffic, which is for Internet to the NAT

gateway, then those traffics would be sent to the internet gateway by using NAT's IP address as the source IP address. (19)

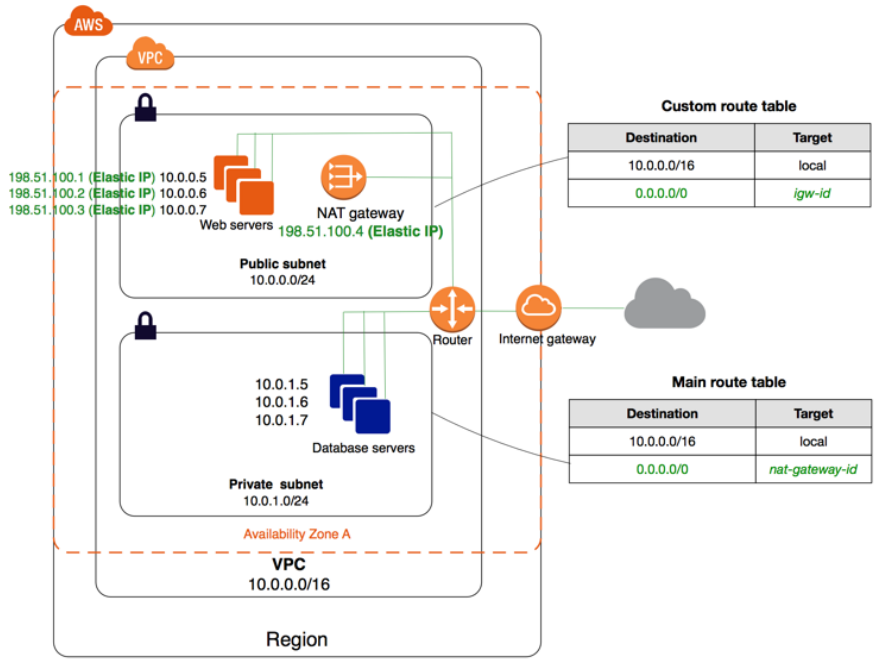

Figure 4. 16

We need to consider the limitation and characteristics of a NAT gateway:

- NAT gateway supports 5 Gbps of bandwidth and automatically scales up to 45 Gbps.
- Each NAT gateway can have only one Elastic IP address that after associating with NAT gateway, it cannot be changed or modified. In order to use a different Elastic IP address for NAT gateway, we need to create a new NAT gateway.
- A NAT gateway supports the following protocols: TCP, UDP, and ICMP.
- The security groups could not be applied to a NAT gateway.
- ACL could be applied to the traffic of NAT gateway in order to control traffics.
- While we create a NAT gateway, an interface would be associated with it automatically and an IP address from the range of subnet would be assigned.
- We would not be able to have access to NAT gateway by using Classic Link.
- A NAT gateway could not be used by resources on the other side of the connection.
- A NAT gateway could handle 55,000 current connections for each destination. (19)

4.4.9.2 NAT Instance

The following figure shows the basic of NAT instance. The main route table is associated with the private subnet and sends the traffic from the instances to the NAT instance in the public subnet. The NAT instance sends the traffic to the Internet gateway for the VPC. The traffic is attributed to the Elastic IP address of the NAT instance. (19)

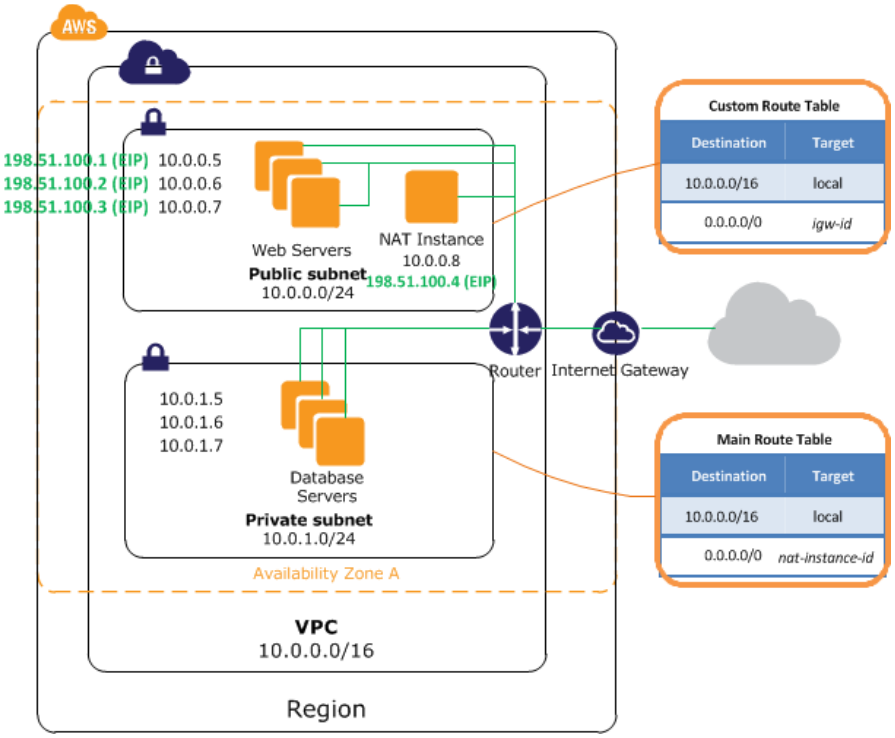

Figure 4. 17

Amazon provides Amazon Linux AMIs which are configured to run as NAT instances. These AMIs include the string “amzn-ami-vpc-nat” in their names.

4.4.10 Connecting to VPC

There are several options to connect to a VPC in amazon;

- Virtual private network (VPN)
- AWS Direct Connect (DX)
- VPC peering connection
- VPC endpoint
- Classic Link (20)

4.4.10.1 VPN

In order to connect to the VPC, we can use a software-based VPN appliance. By launching an AMI from AWS marketplace which is provided by some third-party vendors such as OpenVPN,

this appliance could be installed. This appliance makes a secure encrypted connection to VPC. (Gilchrist & Soni, 2018)

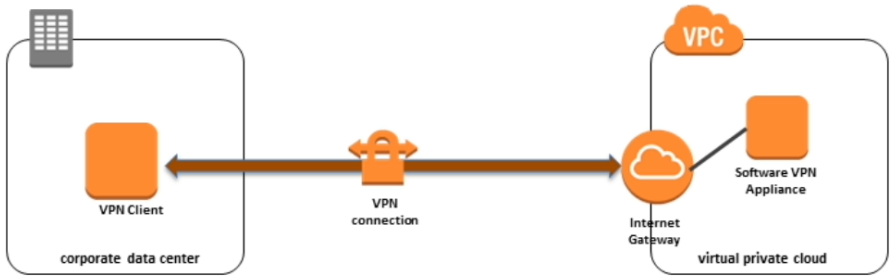

Figure 4. 18

(Gilchrist & Soni, 2018)

4.4.10.2 AWS Direct Connect (DX)

AWS Direct Connect provide a connection to an AWS Direct Connect location over a standard Ethernet or fiber-optic cable. One end of the cables is connected to the router, the other to an AWS Direct Connect router. In this way, actually we are creating virtual interfaces directly to public AWS services or to Amazon VPC, bypassing internet service providers in your network path. An AWS Direct Connect location provides access to AWS in the region with which it is associated. We can use a single connection in a public Region or AWS GovCloud (US) to access public AWS services in all other public regions. (21)

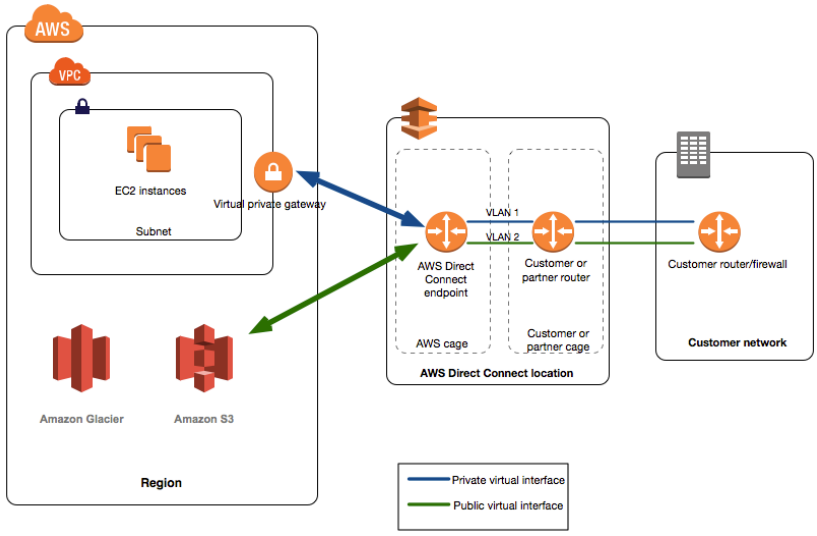

Figure 4. 19

This type of connection has two key components;

- Virtual interface: Virtual interface is used to access to AWS services. A public virtual interface is used to access to a public service such as Amazon S3. A private virtual interface is used to access to VPC
- Connection: A connection in an AWS Direct Connect is used to establish a network connection from the local to an AWS Region. (21)

4.4.10.3 VPC peering connection

A VPC peering connection is a networking connection between two VPCs that is used to route traffic between them by using their private IPv4 addresses or IPv6 addresses. Instances in each VPC can communicate with each other if they locate in the same network. It is possible to create a VPC peering connection between your own VPCs, or with a VPC in another AWS account. The VPCs can locate in different regions. (22)

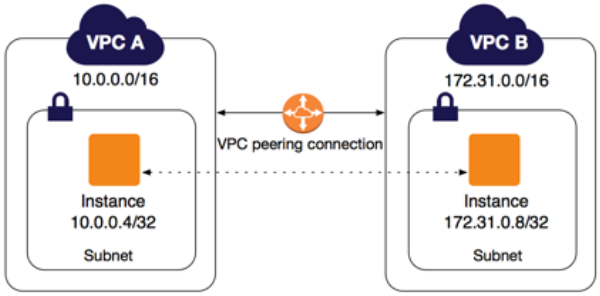

Figure 4. 20

Also, it is possible to establish peering relationships between VPCs across different AWS regions. This allows VPC resources including EC2 instances, Amazon RDS databases and Lambda functions that run in different AWS regions to communicate with each other using private IP addresses, without requiring gateways, VPN connections, or separate network appliances. The traffic remains in the private IP space. All inter-region traffic is encrypted with while there is no single point of failure, or bandwidth bottleneck. Traffic always route on the global AWS backbone, and never goes to the public internet. In this case actually the possibility of each kind of attack is reduced. Inter-Region VPC Peering provides a simple and cost-effective way to share resources between regions or replicate data for geographic redundancy. (22)
In the following, the procedure in order to enable VPC peering connection has been listed;

- The owner of a VPC sends a request to another VPC in order to create the VPC peering connection. But these two VPCs could not have the same IP address range.

- The other VPC accepts this request in order to activate VPC peering connection.
- In order to enable the flow traffic between these two VPCs, the owner of each VPC should add one or more route to route table.
- If it is needed, security group rules should be updated in order to prevent any restriction traffic between these two VPCs. (23)

4.4.10.4 VPC endpoint

This type of connection is a private connection between VPC and another resource in AWS. There are two types of interface; (24)

- Interface VPC endpoint: An interface endpoint is an elastic network interface with a private IP address from the IP address range of the subnet that is considered as an entry point for the traffic which would be routed to a supported service.(25)
- Gateway PVC endpoint: A gateway endpoint is a gateway that is specified as a target for traffics which are routed to a supported AWS service. (25)

In order to create an endpoint service configuration, we need to use the Amazon VPC console or the command line. But it would be better to create one or more load balancers in VPC for the services. (26)

4.4.10.5 Classic Link

This kind of link is used to make a connection between EC2 instances to a VPC in the same account, within the same region. That would be enabled by associating the VPC security groups to an EC2 instance. In this type of connection, there is no need to use a public IPv4 addresses or Elastic IP addresses to enable the communication between instances in these platforms. (27)
There are two steps for using this link. First, we need to enable Classic Link for VPC. By default, it is not active. After that we need to enable this option for all instances in the same region of VPC. Linking the instance includes selecting security groups from the VPC to associate with the EC2-Classic instance. The EC2-Classic instance does not lose its private IP address when linked to the VPC. (28)

4.5 Securing VPC

In this section, we will discuss about network ACL which can add a more security layer to our VPC.

4.5.1 Network ACL

A network access control list (ACL) is an optional layer of security for a VPC that causes VPC acts as a firewall for controlling traffic in and out of one or more subnets. The rules of ACL are similar to the security groups. (29)

In the following, some basics of ACL has been explained;

- Each VPC automatically has a modifiable default network ACL. By default, it allows all inbound and outbound IPv4 traffic.
- We can create a custom network ACL and associate it with a subnet. By default, each custom network ACL denies all inbound and outbound traffics until the rules are added.
- Each subnet in the VPC must be associated with a network ACL. If we don't explicitly associate a subnet with a network ACL, the subnet is automatically associated with the default network ACL.
- We can associate a network ACL with multiple subnets. However, a subnet can be associated only one network ACL at a time. When we associate a network ACL with a subnet, the previous association is removed.
- A network ACL contains a numbered list of rules. The rules evaluated in order, starting with the lowest numbered rule, to determine whether traffic is allowed in or out of any subnet associated with the network ACL.
- A network ACL has separate inbound and outbound rules, and each rule can either allow or deny traffics.
- Network ACLs are stateless, which means that responses to allowed inbound traffics are subject to the rules for outbound traffics. (30)

4.5.2 High Availability architecture

In order to make our environment high available, instances should be deployed into multiple availability zones and all the requests should be shared among them by load balancers. (Gilchrist & Soni, 2018)

4.5.2.1 Availability zones

AWS has an infrastructure in the most of countries such as Europe, Asia, Australia, Canada and Brazil. Each country is divided into several areas where are separated from each other and called availability zones. These zones inside a region are connected to each other with very low latency network connections. Although VPCs should be located in one region, but we can spread them

between several zones. In this case, if one of the zones face any outage issue, the other VPCs in other zones respond the requests.

The following diagram shows a highly available VPC architecture, in which we have created subnets for our instances, in two different AZs. (Gilchrist & Soni, 2018)

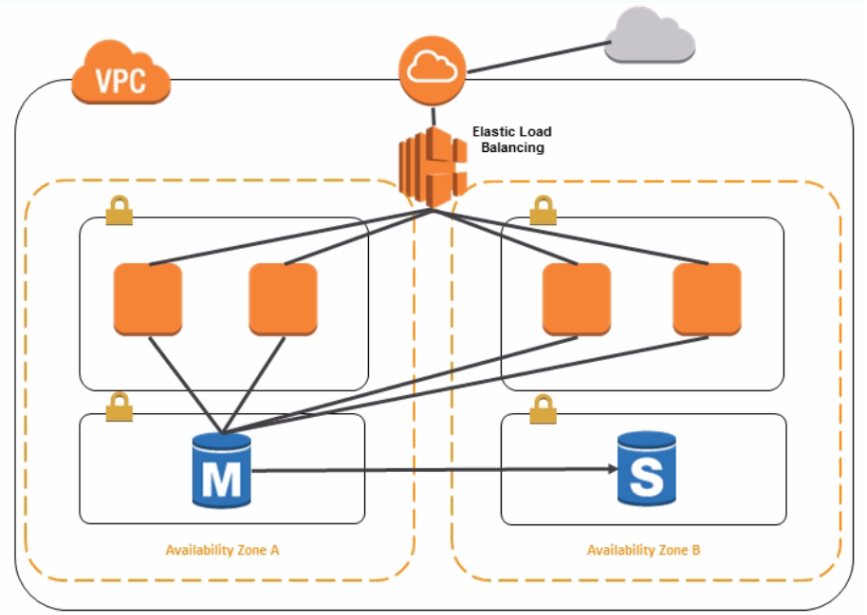

Figure 4. 21

(Gilchrist & Soni, 2018)

4.5.2.2 Load Balancer

There are three types of load balancers in AWS, Application Load balancer, which is useful to route traffic for Layer 7, Network and Classic load balancer which is useful for traffic layer 4. (32)

4.5.2.2.1 Application Load Balancer

This type of load balancer routes traffic based on the application layer and route requests to one or more ports of a container in a cluster. (33)

4.5.2.2.2 Network Load Balancer

This type of load balancer works based on layer 4 and attempts to open a TCP connection to the destination machine. (33)

4.5.2.2.3 Classic Load Balancer

A Classic Load Balancer routes traffic based on the layer 4 or Layer 7. (33)

4.6 Auto Scaling

Amazon EC2 Auto Scaling creates a possibility to make sure enough quantity of EC2 instances is available to handle the load for an application. We need to specify the minimum and maximum number of instances for each auto-scaling group. To maintain our high availability, we should specify a minimum of not less than two. Each auto scaling group needs to have a launch configuration. This is simply all of the attributes of the EC2s that the auto scaling group will launch, including the AMI, instance size and type, security group, and so on. The way that auto-scaling knows when it is time to launch or terminate instances, is through the CloudWatch monitoring service. We can configure alarms based upon default metrics, such as average CPU utilization, for the instances in the end or average response latency measured at the ELB. Auto-scaling launches and terminates instances, when notified by CloudWatch that an alarm has been triggered**.** (Gilchrist & Soni, 2018)

4.7 Amazon CloudFront

The Amazon CloudFront service is a CDN service for low latency content delivery (static or streaming content). For instance, copies of S3 objects can be distributed and cached at multiple locations over the world, by using this kind of service.

4.8 Amazon Glacier

Amazon Glacier is low-cost storage service that is usually used for archiving and backup, but we need to mention that retrieval time for data from Glacier is up to several hours. Other AWS Storage services are available including Amazon Storage Gateway (enables integration between on-premise environments and AWS storage infrastructures) and AWS Import/Export service (which uses portable storage devices to enable movement of large amounts of data into and out of the AWS cloud environment).

4.9 Amazon RDS

Amazon Relational Database Service **(**Amazon RDS**)** provides an easy way to set up, operate, and scale a relational database in the cloud. Database options available from AWS include MySQL, Oracle, SQL Server, PostgreSQL, and Amazon Aurora.

4.10 Amazon Route 53

Amazon Route 53 is a highly scalable DNS service that would be useful to manage DNS records by creating a hosted zone for every domain.

4.11 AWS Identity and Access Management

AWS Identity and Access Management **(**IAM**)** provide us control access to AWS services and resources. We can create users and groups with unique security credentials and manage permissions for each of these users. You can also define IAM roles so that an application can securely make API calls without creating and distributing the AWS credentials. IAM is natively integrated into AWS Services.

4.12 Amazon CloudWatch

CloudWatch is a monitoring service for your AWS resources. It provides us possibility to monitor data, set alarms, troubleshoot problems, and take actions based on each type of issue.

4.13 Amazon free service

All free services of Amazon are listed in https://aws.amazon.com/free/. Some of the services are free only for first year such as Amazon EC2, Elastic Load Balancing and RDS and some of other services would be free after first year such as 25G capacity in Amazon DB. (Curator, 2017)

5. Comparison of cloud service providers

Here we want to have a comparison between the three largest cloud service providers over the world, Azure with 16% market share, Google with 8% and AWS with 33% sits at the top. (68)

5.1 Azure Vs GCP: Key similarities and Differences

Generally speaking, the purpose of Azure is for running applications and handling storage for the enterprise environment and has the possibility to connect to on-premises Windows-based system. On the other hand, Google has good services to host a number of different systems and is designed to handle applications and enterprise environments. The main concentration of Google is public cloud computing while Azure focuses on those ones who want to interoperate with their own data centers where lots of Windows servers are running. It means that Azure is one of the hybrid cloud option. (68)

Regarding similarity between these two cloud options, it should be mentioned that users need to sign up by a Microsoft account in order to use Microsoft Azure services. After they have completed the required process, they would be able to launch any services under their accounts by considering Azure limitations and all charges goes to the specific account. We have the same procedure in Google as well, users need to use Google account in order to use Google services but instead of charging per account, Google charged based on the projects. In this way, users could be able to create several projects under the same account. One of the benefits of this model is allowing users to create project space for separate divisions or groups inside a company. Another advantage is for testing purpose. (68)

5.1.1 Resource Management interface

Azure and Google provide a command line interface in order to interact with resources and services. Google uses Cloud SDK while Azure uses Windows command line and each of them is cross-platform with binaries available for Windows, Linux and Mac OS. Moreover, in GCP we could use Cloud SDK in our browser by using Google Cloud shell. Also, both providers have web-based console to create, manage and monitor the resources. (68)

5.1.2 Pricing Process

In Azure environment, users would be charged based on the type of products and the range of cost would be between 0.99$ to 0.14$ per hour while in AWS and GCP the factor would be based on per GB of RAM. Google follows to-the-minute pricing model. Actually, the platform has pay-as-you go pricing, billing to per second of usage, although GCP offers discounts for long-term usage which is started after the first month. (68)

5.1.3 What Microsoft Azure is

Azure is a cloud service platform which is designed and built by Microsoft and launched in 2010. It starts its competition with AWS, which has been launched 4 years sooner by providing the same services such as compute, storage, networking, developer tools and other functionalities which provide the possibility for organizations to scale and grow their business. Azure provides services under all three categories, IaaS, PaaS and SaaS. All services would be useful for developers and software employees to create, deploy and manage services and applications through the cloud. Also, it offers a wide range of integrated cloud services and functionalities such as analytics, computing, networking, database, storage, mobile and web applications that integrate with enterprise environments in order to achieve efficiency and scalability. (68)

5.1.4 What Google cloud is

Google cloud is a cloud computing platform which has been launched in 2008. It was written in Java, C++, Python and ruby. Also, it provides different services such as Iaas, Paas and Server-less platform. Google cloud is categorized into different platforms such as Google App Engine, Google Cloud Datastore, Google cloud Storage and Google cloud SQL. GCP provides a high-level computing, storage, networking and databases. Regarding networking, there are several options such as virtual private cloud, cloud CDN, cloud DNS, load balancing and other features. (68)

5.2 AWS vs Azure

Here we compare AWS and Azure from different aspects such as storage, computing, security, network, database, pricing and so on.

5.2.1 Storage

Aws provides temporary Elastic Block storage that would be associated with an instance when a machine is started and terminates when it ends. While Azure offers Azure blob storage, Azure core storage services and table and file storage. Also, it has the possibility to transfer huge data through Import/Export site. Regarding backup and archiving, user could use utilize Azure backup and site recovery. (69)

5.2.2 Computing

Both AWS and Azure provide competitive computing power but there is a thin line of difference. Azure focuses on VM where users could be able to choose VHDS for their VM. Also, it would be possible to determine memory size, cores or pre-configured one from Microsoft. On the other hand, AWS uses EC2 that provides scalable computing on-demand and would be customized for

several options. Moreover, we could choose the memory, power and number of VMs. Amazon web service provides EC2-related service such as EC2 container, auto-scaling, AWS Lambda and Elastic Beanstalk app deployment. (69)

5.2.3 Security

Security has been one of the big headaches for organizations, especially when they move to the cloud solutions. Security in cloud is based on role-based access control that manages how users would be controlled in order to access resources. Azure uses AD that is a version of RBAC whose capabilities exceed those of AWS's RBAC. Indeed, AWS encourages users to use Azure Active Directory. But it does not mean that AWS does not have any other security tools. It has sophisticated security tools such as DDoS protection and guard security which is closely to AD feature. (69)

5.2.4 Database

AWS and Azure both provide database service with rational database and NoSQL. Amazon offers RDS (rational database service) that has more controls and options to the users comparing to Azure equivalent. Regarding NoSQL, Amazon provides Dynamic DB that can be used with RDS to allow stronger cloud environment for large data volume. While Azure, NoSQL is integrated with big data. (69)

5.2.5 Networking

Both Azure and AWS provide the possibility to connect to our on-set premise system. Within Amazon, users would be able to create isolated networks by using virtual private cloud. Also, they would be able to create subnets, rout tables, network gateway and private IP address. (69)

5.2.6 Pricing

AWS and Azure both use pay as you go method, but Azure charge users based on minute-by-minute while AWS charging is based on the second. Also, both offer free trail with limited usage for new users. (69)

5.3 AWS vs Google

As we already mention about AWS, it is one of the public cloud providers, but we need to compare these two providers together in order to make a decision in rational way. (70)

5.3.1 Market share

As the statistics shows AWS had revenues of 6.2$ billion and its market share in IaaS was about 31%, comparing to Azure which sits in the second place with 18% market share and Google in the third place with 8% market share in Q2 of 2018. But it should be considered that AWS

launched its public cloud platform in 2006 while Google launched its PaaS Google App engine in 2008 and did not enter to the IaaS market until 2010. Another aspect is that during launching Google, it has 108% grew while AWS had 48%. (70)

5.3.2 Pricing

Regarding pricing between these two cloud providers, each has a different plan. Over one year, Google cloud is 28% cheaper than AWS while AWS offers 35% cheaper over 3 years between Google compute Engine and EC2 of AWS. Generally, Google provides cost reduction on over long-term use while there is no upfront cost to achieve those discounts for AWS. (70)

5.3.3 Feature and service

The variety services in Google are about 50 while it cannot be compared with AWS with more than 200 service. Those additional services are important for large enterprises since they need exact requirement while Google offers a core range of services which meet the needs for most business. VMs in Google are most customized since there are no predefined instances while in AWS most of the instances have been predefined to meet the needs of most business. (70)

5.3.4 Global Reach

Generally, AWS has better score in terms of global reach by providing more data center around the world. As of sep2018, Google had 17 regions, 52 zones and over 100 points of presence in 35 countries while it cannot be compared with AWS with 18 region, 55 availability zone, and one local region and had customers in 190 countries. (70)

5.3.5 Free trial

Both of them have 12 months free trail. AWS free trail includes 750 hours compute on EC2, 5GB storage of S3, 750 hours per month on Amazon RDS while google offers a 300$ credit which is valid for 12 months. In addition, we need to consider always free services in Google that is more generous than those are being offered by AWS. It includes 28 instances hours/5GB storage on Google App Engine, 1 GB storage on its NoSQL database, 1 f1-micro instance per month with a 30GB HDD, 5GB storage per month. (70)

6. References

1. Retrieved from https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/virtualization_types.html

2. Retrieved from https://aws.amazon.com/about-aws/whats-new/2015/06/introducing-m4-instances-and-lower-amazon-ec2-instance-prices/

3. Retrieved from https://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/Storage.html

4. Retrieved from https://aws.amazon.com/ec2/instance-types/t2/

5. Retrieved from https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ec2-security-groups.html

6. Retrieved from https://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/accelerated-computing-instances.html

7. Retrieved from https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/compute-optimized-instances.html

8. Retrieved from https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/memory-optimized-instances.html

9. Retrieved from https://aws.amazon.com/ec2/instance-types/r5

10. Retrieved from https://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/storage-optimized-instances.html

15. Retrieved from https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/AMIs.html

16. Retrieved from https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ec2-key-pairs.html

17. Retrieved from https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/using-instance-addressing.html

18. Retrieved from https://docs.aws.amazon.com/vpc/latest/userguide/VPC_Route_Tables.html

19. Retrieved from https://docs.aws.amazon.com/vpc/latest/userguide/vpc-nat.html

20. Retrieved from https://aws.amazon.com/premiumsupport/knowledge-center/connect-vpc/

21. Retrieved from https://docs.aws.amazon.com/directconnect/latest/UserGuide/Welcome.html

22. Retrieved from https://docs.aws.amazon.com/vpc/latest/peering/what-is-vpc-peering.html

23. Retrieved from https://docs.aws.amazon.com/vpc/latest/peering/vpc-peering-basics.html

24. Retrieved from https://aws.amazon.com/premiumsupport/knowledge-center/connect-vpc/

25. Retrieved from https://docs.aws.amazon.com/vpc/latest/userguide/vpc-endpoints.html

26. Retrieved from https://docs.aws.amazon.com/vpc/latest/userguide/endpoint-service.html#create-endpoint-service

27. Retrieved from https://docs.aws.amazon.com/vpc/latest/userguide/vpc-classiclink.html

28. Retrieved from https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/vpc-classiclink.html#classiclink-basics

29. Retrieved from https://docs.aws.amazon.com/vpc/latest/userguide/vpc-network-acls.html

30. Retrieved from https://docs.aws.amazon.com/vpc/latest/userguide/vpc-network-acls.html#nacl-basics

31. Retrieved from https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/using-regions-availability-zones.html

32. Retrieved from https://docs.aws.amazon.com/AmazonECS/latest/developerguide/load-balancer-types.html

33. Retrieved from https://docs.aws.amazon.com/AmazonECS/latest/developerguide/load-balancer-types.html

34. Retrieved from https://docs.aws.amazon.com/autoscaling/ec2/userguide/what-is-amazon-ec2-auto-scaling.html

41. Retrieved from https://docs.microsoft.com/en-us/azure/azure-resource-manager/resource-group-overview

42. Retrieved from https://azure.microsoft.com/en-ca/global-infrastructure/regions/

43. Retrieved from https://www.mulesoft.com/resources/api/what-is-rest-api-design

44. Retrieved from https://docs.microsoft.com/en-us/azure/azure-resource-manager/resource-group-overview

45. Retrieved from https://docs.microsoft.com/en-us/azure/azure-resource-manager/resource-group-using-tags

46. Retrieved from https://docs.microsoft.com/en-us/azure/azure-resource-manager/resource-group-lock-resources

47. Retrieved from https://docs.microsoft.com/en-us/azure/azure-resource-manager/template-deployment-overview

48. Retrieved from https://docs.microsoft.com/en-us/azure/active-directory/hybrid/whatis-azure-ad-connect

49. Retrieved from https://docs.microsoft.com/en-us/azure/virtual-network/virtual-networks-overview

50. Retrieved from https://docs.microsoft.com/en-us/azure/virtual-network/security-overview

51. Retrieved from https://docs.microsoft.com/en-us/azure/virtual-network/service-tags-overview

52. Retrieved from https://docs.microsoft.com/en-us/azure/load-balancer/load-balancer-overview

53. Retrieved from https://docs.microsoft.com/en-us/azure/vpn-gateway/vpn-gateway-about-vpngateways?toc=%2fazure%2fvirtual-network%2ftoc.json#P2S

54. Retrieved from https://docs.microsoft.com/en-us/azure/virtual-network/virtual-networks-udr-overview#user-defined

55. Retrieved from https://docs.microsoft.com/en-us/azure/virtual-network/virtual-networks-udr-overview#user-defined

56. Retrieved from https://docs.microsoft.com/en-us/azure/vpn-gateway/vpn-gateway-bgp-overview?toc=%2fazure%2fvirtual-network%2ftoc.json

57. Retrieved from https://docs.microsoft.com/en-us/azure/storage/common/storage-account-overview

58. Retrieved from https://cloudacademy.com/blog/azure-storage-service-overview/

59. Retrieved from https://docs.microsoft.com/en-us/azure/storage/common/storage-account-overview

60. Retrieved from https://docs.microsoft.com/en-us/azure/storage/common/storage-account-keys-manage

67. Retrieved from https://aws.amazon.com/products/

68. Retrieved from https://www.eweek.com/cloud/microsoft-azure-vs-google-cloud-platform

69. Retrieved from https://dashbird.io/blog/aws-vs-azure/

70. Retrieved from https://www.cloudhealthtech.com/blog/google-cloud-vs-aws

Blokland, K., Mengerink, J., & Pol, M. (2013). *Testing cloud services.* Rocky Nook.

Curator, T. K. (2017). *AWS Environments with AWS Lambda.* Packt Publishing 2017.

Geewax, J. (2018). *Google Platform in action.* Manning Publications.

Gilchrist, W., & Soni, M. (2018). *Designing AWS Environments.* Packt Publishing,.

Jamsa. (March 2012). *Cloud Computing.* Jones & Bartlett Learning.

Modi, R., Damaschke, J.-H., Klaffenbach, F., & Michalski, O. (2018). *Deployment of Microsoft Azure Cloud Solutions.* Packt Publishing.

Patel, A. (n.d.). Retrieved from https://medium.com/awesome-cloud/aws-difference-between-ebs-and-instance-store-f030c4407387

Shah, A., & Sarkar, A. (2015). *Learning AWS.* Packt Publishing.

Shah, A., & Sarkar, A. (n.d.). *Learning AWS.*

Zaal, S. (2018). *Architecting Microsoft Azure Solutions - Exam Guide 70-535.* Packt Publishing.