ISM Original Work

AWS Cloud Architecture for Cloud Computing

Aloksai Choudari

December 2019

ISM Original Work

For my Original Work this year in ISM, I decided to create the cloud diagrams I saw through my reading and a cloud architecture diagram of my own using a website called CloudCraft that allows individuals and businesses to plan, create, and implement a cloud for company and/or personal use. The website helps visualize an environment that utilizes applications from one of the largest cloud computing platforms in the world, Amazon Web Services (AWS), and displays prices for each of the AWS products that are used throughout the design. The website presents a blueprint through 3D graphics to emphasize the sizeable demonstration of projects, which can be expanded to multiple pages in the site. Along with this, I also wanted to write explanations of the specific products from AWS and how they are used throughout the designs I created.

In order to create a cloud design of my own, I browsed through a few examples to comprehend my plan of action and help me depict what I was going to create. I looked through many different examples spanning from small companies to large companies, such as Apple and Verizon, which gave me a broad range for my project conception. The Apple applications I looked at showed me the actual code that is executed behind the design through the cloud process, which was not too helpful in terms of what I was going for, while the Verizon cloud example I looked at showed the physical design of the cloud, which was what I was trying to design for my Original Work. Although I was able to look through these designs, I was not allowed to include any pictures of them on my Original Work due to security reasons of the companies. One of the most crucial challenges I faced through this process was that all of the blueprints that I looked at were, in fact, way too large and far beyond my current knowledge in

the subject. In order to create my own design, I had to minimize the scope of my design outlook substantially. Another challenge I faced was with my understanding of the AWS products being used by CloudCraft. Up until the point that I started my Original Work, I knew there were many applications in AWS, but I did not know what each application was best fit for, how they were supposed to be used, where I should be placing them in my design, and most importantly, why was the best place to position each product at their respective regions.

Eventually, I ended up splitting my Original Work into three parts: writing the definitions and explanations for each of the products being used throughout the blueprints I create on CloudCraft, create the cloud examples that I look at, and then create a design of my own using all of my previous research. The first part of my project was all about researching the AWS products that were provided in CloudCraft and finding out how I was to use them in my design. After that, I used the examples that I looked at initially and references from the research assessments I did prior to this time to create a few sample cloud designs. Finally, I created a design from complete scratch and wrote a brief explanation as to what I was going for in my creation, why I used certain products instead of others, and how I was going to expand on my diagram in my Original Work to eventually create something greater for my Final Product by the end of the year.

AWS Basic Applications

Auto Scaling - Based on my research, there is Auto Scaling for many different resources in AWS. The overall point of AWS Auto Scaling is to monitor applications and automatically adjust capacity to maintain steady and proper performance of all services. The specific Auto Scaling that was mentioned in CloudCraft was referring to EC2, which is what I researched more into depth. Amazon EC2 Auto Scaling, specifically, ensures that the correct number of Amazon EC2 instances are available to handle the load for application purposes. The minimum and maximum numbers of instances can be set through the EC2 Auto Scaling, which can be launched and terminated by this service itself. Altogether the EC2 Auto Scaling is beneficial as it can terminate instances whenever they are harmful, and it is a better source of availability and cost management for a business or individual.

EC2 - Amazon EC2 stands for Amazon Elastic Compute Cloud, which provides scalable computing capacity in the AWS cloud. This eliminates the need for a business or an individual to buy hardware up front, so applications in the cloud can develop and deploy faster. Amazon EC2 is also used to provide virtual servers from the cloud known as instances. These instances are launched from the Amazon Machine Image (AMI), which is a template that contains the actual software configuration behind the AMI displays. Overall, Amazon EC2 is a web service that provides secure and resizable compute capacity within the cloud that provides complete control over computing resources.

Lambda - AWS Lambda is used to run the code that one creates in order to minimize server provisions and management as well as reduce computing time. In order to successfully operate with Lambda, the proper code just has to be uploaded and the service takes care of the rest of the

administration. The code provided to run within the Lambda service is interconnected with other AWS services in order to facilitate more efficient responses to specific events that take place in the cloud network. The primary benefit of the Lambda service is to allow serverless computing possible to speeden transition times in the cloud.

Availability Zone - Availability Zones are the locations that contain available data center facilities that house all of the AWS cloud computing resources and provide them to users whenever needed. These availability zones are isolated from other availability zones in different regions in order to prevent failures and crashes in each of the zones. Whenever an instance is launched from the AMI, users can select specific availability zones to communicate the instances with. These availability zones are reached by specific region codes that are special to each of the regions where the availability zones exist.

EBS - Amazon EBS stands for Amazon Elastic Block Store, which is a block storage service that is designed to work along with the Amazon EC2 instances. Basically, EBS acts like a block device, such as a hard drive, that can be attached to EC2 instances, acting as an independent storage volume, and eventually increase the availability and reliability of storage volumes. EBS volumes are only allowed to be used with certain instances that were made in the specific Availability Zones they came from. The most beneficial part of EBS volumes is that it enables the user to increase the amount of storage provisions without disrupting any critical workloads.

S3 - Amazon S3 stands for Amazon Simple Storage Service, which is, altogether, a storage for the internet. S3 allows users to store, protect, and retrieve data ranging from many cases, such as websites, IoT devices, mobile apps, and enterprise apps. S3 is designed for users to make web-scale computing easier, which is used for delivering larger cloud computing resources

within a certain company. Overall, S3 is a simple application, but a very durable and scalable service as well.

EFS - Amazon EFS stands for Amazon Elastic File System, which is a Network File System (NFS) service that alleviates the accommodation of shared access to files in a business. The primary role of EFS is to provide shared access to a numerous amount of EC2 instances. EFS eliminates the need for constant provisioning and managing of capacity growth. EFS is a regional service that is based on certain Availability Zones and varies according to the user in those specific zones. Altogether, EFS supports the configuration of instances and can be used as a data source for workloads running on multiple instances.

CloudFront - Amazon CloudFront is a Content Delivery Network (CDN) service that speeds up the distribution of web content to various users. A CDN, in general, is a network of distributed servers that deliver web pages and other content to users based on geographic location and origin of server/web page. CloudFront specifically is integrated with AWS in order to work with other services S3, EC2, and Elastic Load Balancing. Overall, CloudFront is known for the globally distributed network that it provides and the sustainable distribution over a large amount of users

ELB - ELB stands for Elastic Load Balancing, which basically refers to the three types of load balancers: Application Load Balancer, Network Load Balancer, and Classic Load Balancer. The Application Load Balancer is primarily used for load balancing of HyperText Transfer Protocol and the extension of that and is supposed to correctly guide the traffic of incoming content requests into the Virtual Private Cloud (VPC). The Network Load Balancer is meant to route the traffic to specific targets rather than traveling to violate traffic patterns. Classic Load Balancer is a very basic Load Balancer that works for EC2 instances and operates at different levels in the

network. Overall, ELB distributes incoming traffic to multiple different targets in multiple Availability Zones.

Route 53 - Amazon Route 53 is a Domain Name System (DNS) web service that is highly advanced compared to similar services. A DNS is a naming database where domain names on the internet are translated to IP addresses, which is the way computers communicate and provide information through. Specifically, Route 53 uses special infrastructure that can connect and route with other AWS applications, and can use special feature to conduct traffic and health checks of the cloud.

VPC Gateway - The Amazon Virtual Private Cloud (VPC) gives the user the ability to launch other AWS applications into a virtual network. Basically to visualize the bigger picture, the user is on the outside of the VPC Gateway, and can send and receive certain information through this gateway. Behind the VPC Gateway is where all of the AWS resources are stored for user necessities. VPC endpoints in the gateway allow the user to create connections with other AWS services. These endpoints are also virtual devices or instances.

RDS - Amazon RDS stands for Amazon Relational Database Servers, which are used to run smooth operations in the cloud. Relational Databases stores data information in tables, which makes it easy to compare information through the columns. Amazon RDS, closely, makes it easy to install and run database software without constant maintenance. The database engines that run on RDS can defer from many different choices and are up to the users choice to decide which one of the engines to use in the cloud.

DynamoDB - AWS DynamoDB is a database service that is used for fast performance in the cloud. With DynamoDB, the user can store and retrieve data that they desire at any traffic levels

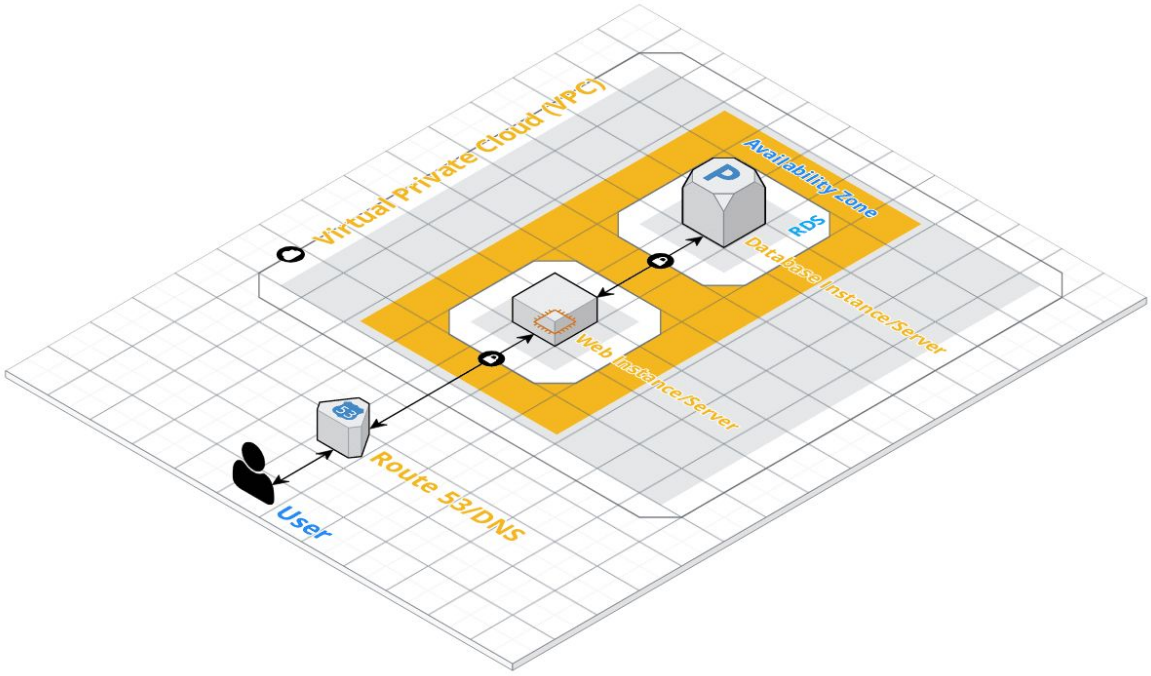
of requests. DynamoDB is for internet-scale applications and can deliver millisecond performances with scalability. DynamoDB is also great for security features of the cloud as it uses encryption attributes, which eliminates the constancy for protecting sensitive data.

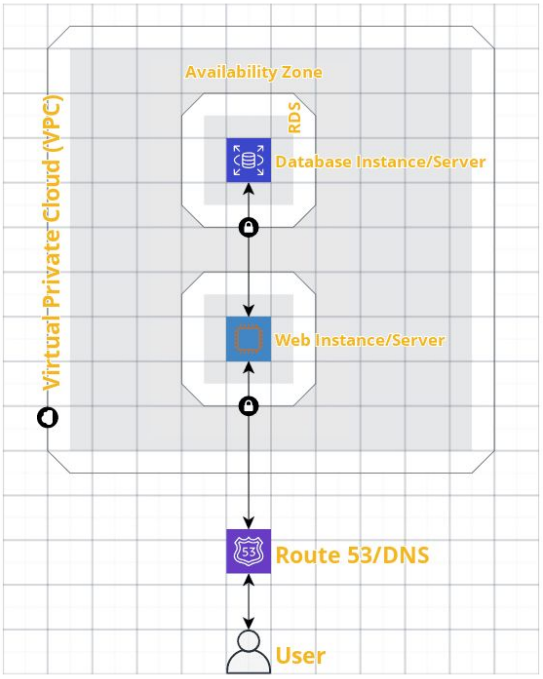
ElastiCache - Amazon ElastiCache is a web service that makes data storage and caching easy. Amazon ElastiCache works with Redis, an in-memory data structure engine that supports abstract information to manage a cache environment, and any existing applications in the cloud that use Redis as well can use ElastiCache with no modifications and with ease. Overall, ElastiCache is an important storage application that provides efficiency along with security.

API Gateway - The main purpose of the Application Programming Interface (API) Gateway is to secure data from any outside access or harmful threats. API makes it easier for developers to monitor the safety of incoming and outgoing data in the cloud. API basically acts as a security guard for resources that monitors how data is being accessed from various services. API access can only be gained through the security measure known as AWS Identity and Access Management (IAM). API Gateway can also run at the same time with multiple different APIs.

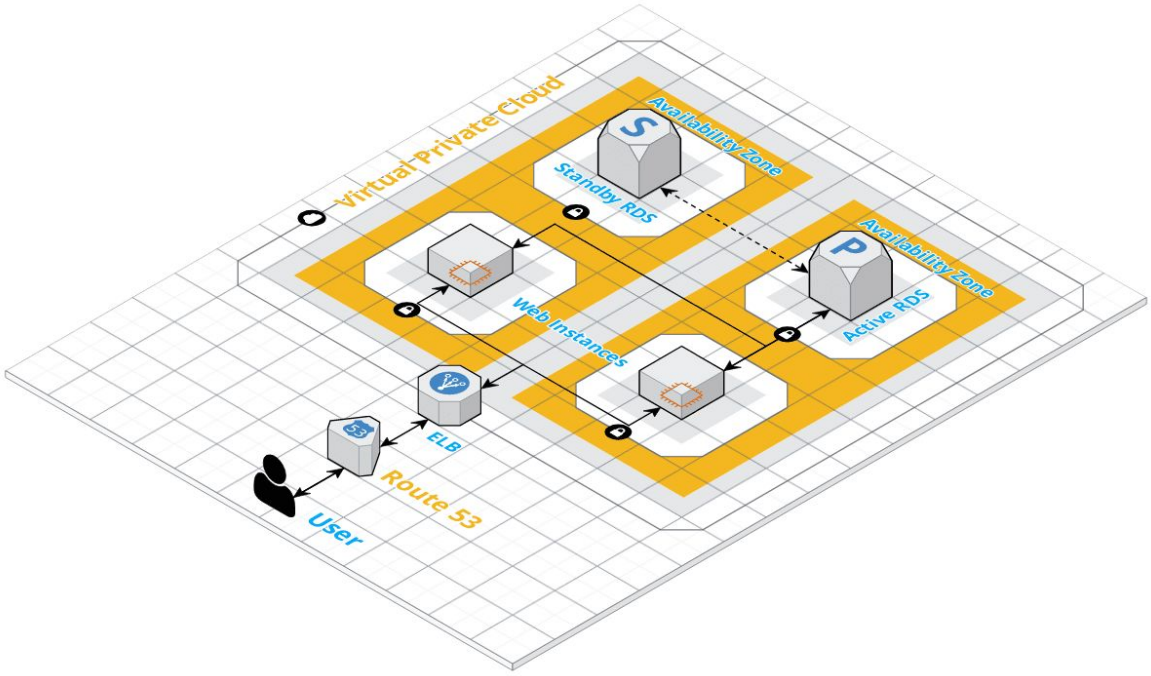
Kinesis Stream - AWS Kinesis Data Streams are used to collect data records or read data from a data stream. Kinesis Data Streams could also use the Kinesis Library to run EC2 instances in the VPC. Along with this, Kinesis Streams ensure that data is never lost and create a map to properly stream certain data applications.

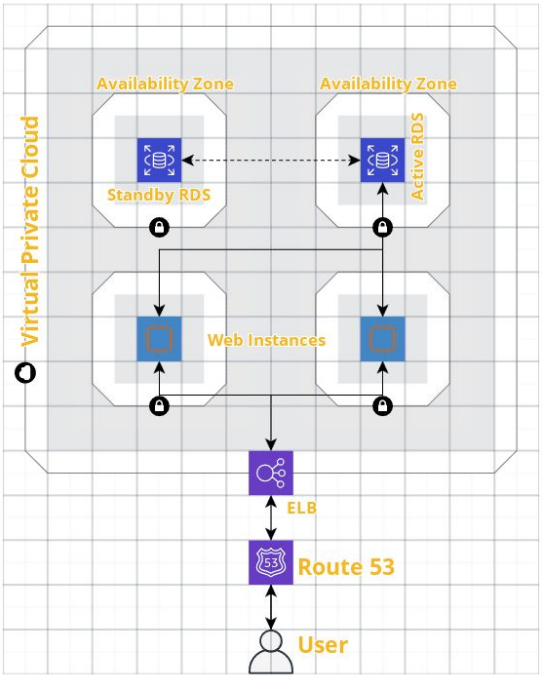
Simple Web Application Architecture



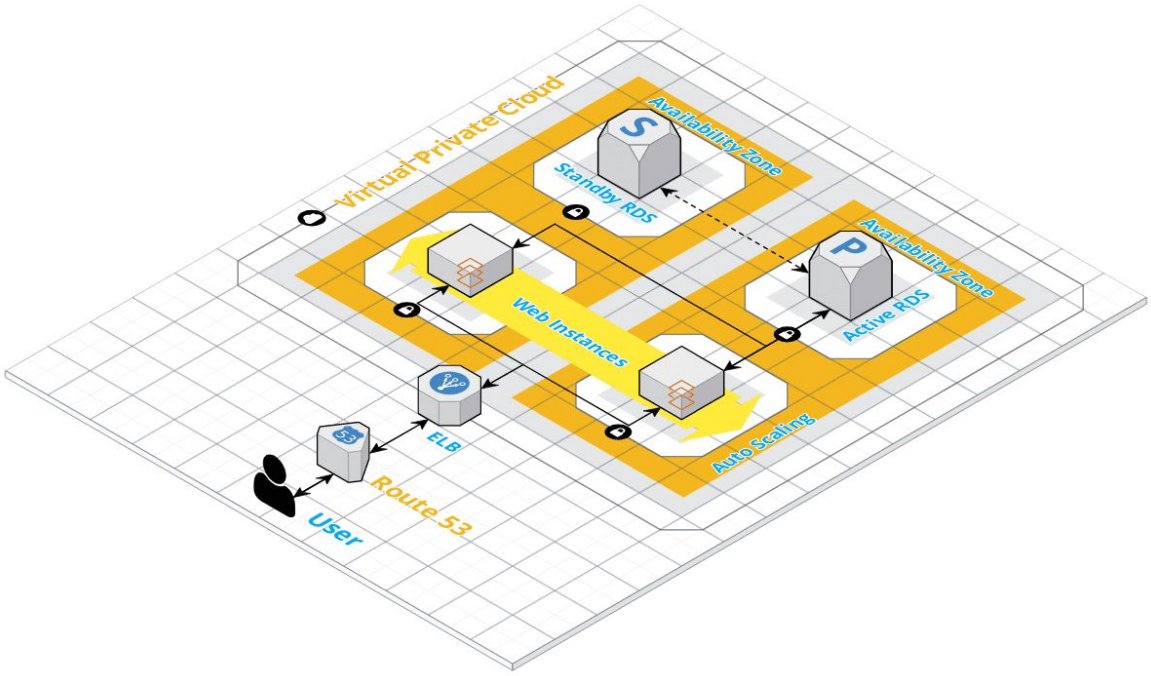


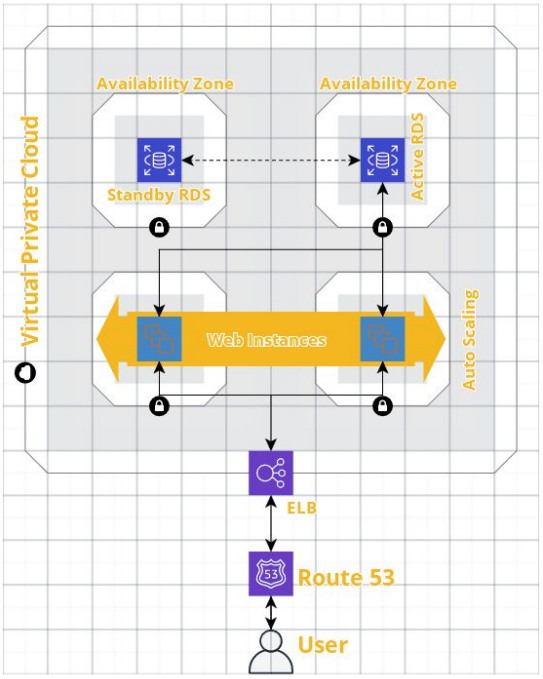
Web Application Architecture with Redundancy



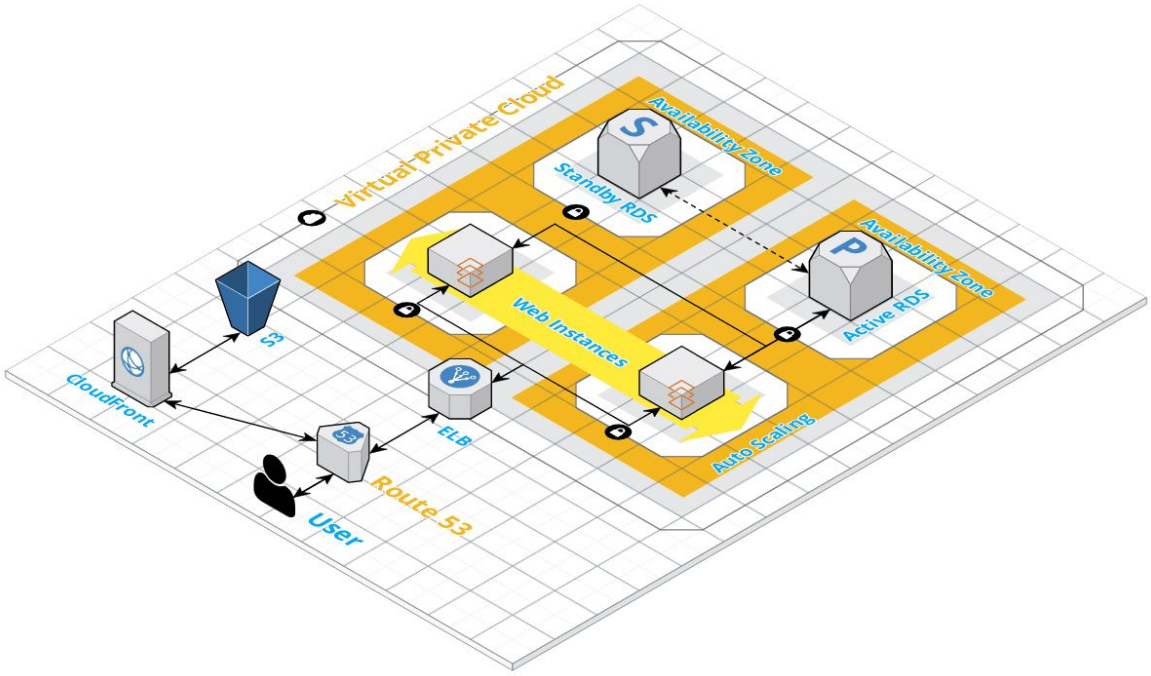


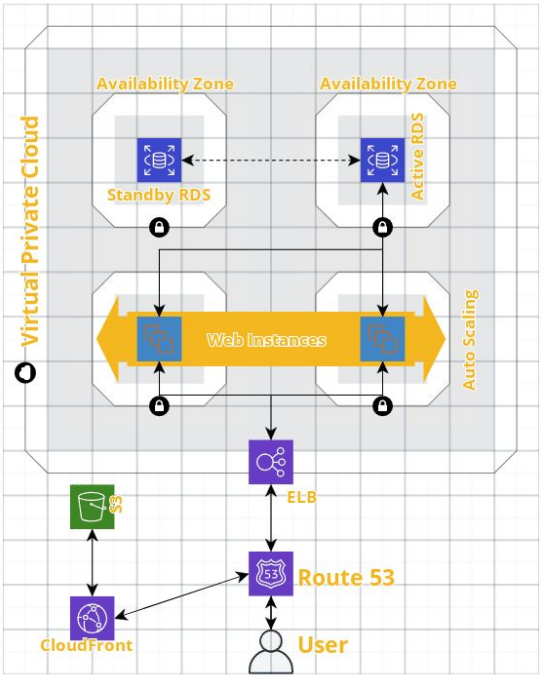
Web Application Architecture with Auto Scaling



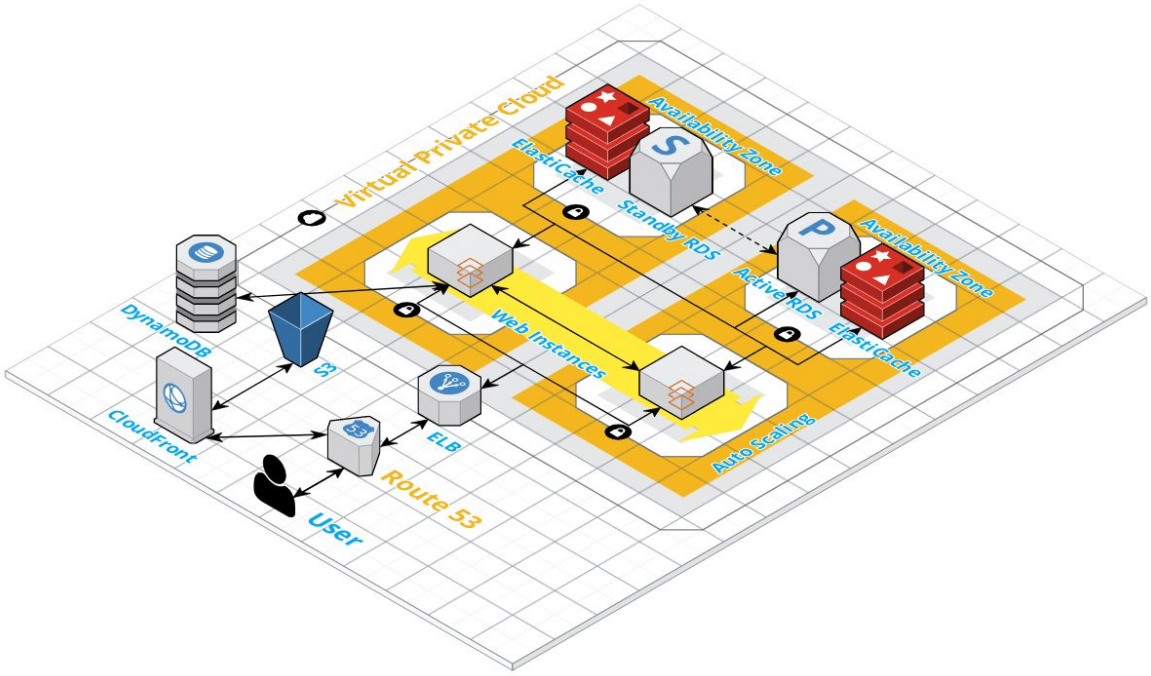


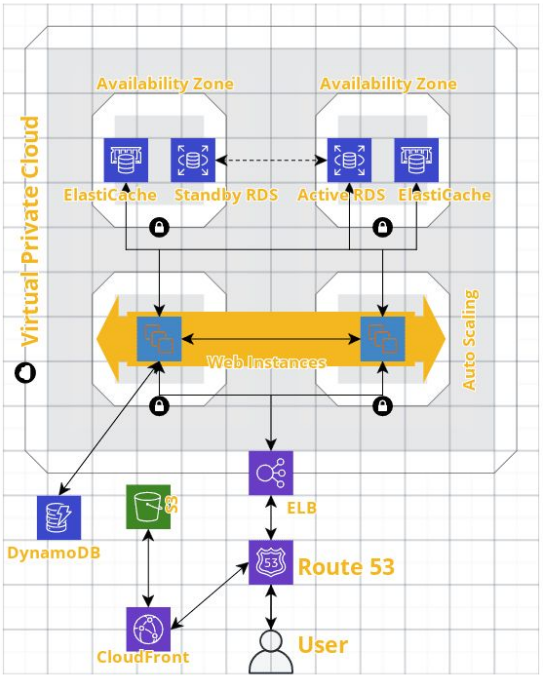
Web Application Architecture with Amazon S3 and CloudFront



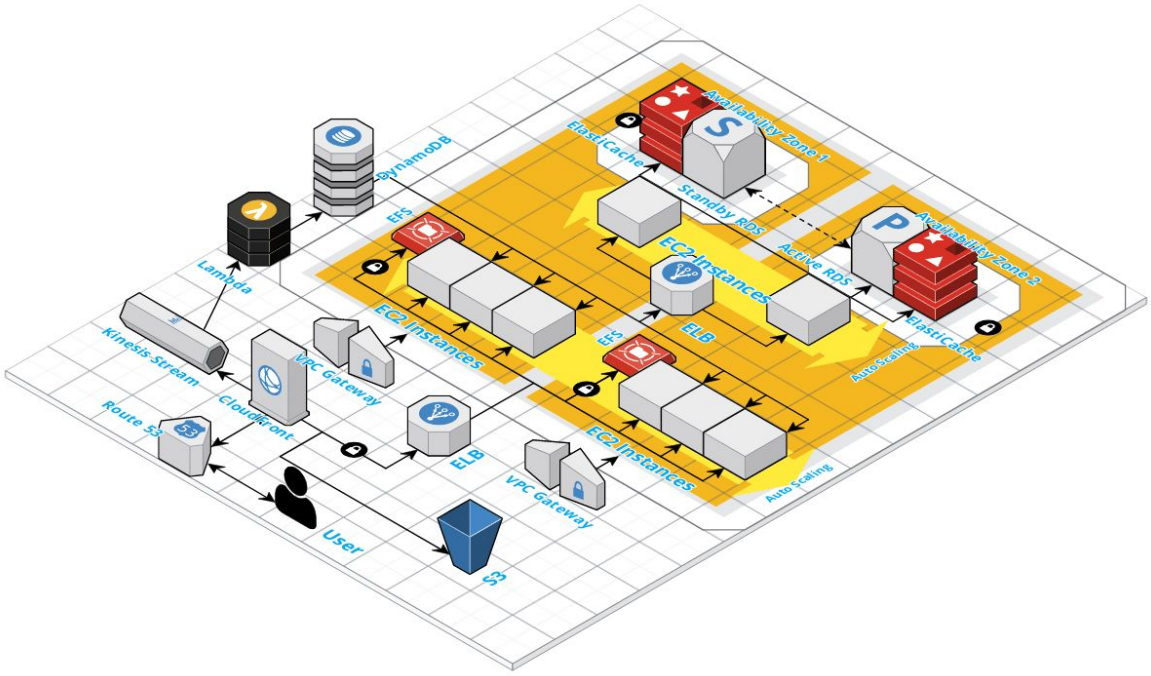


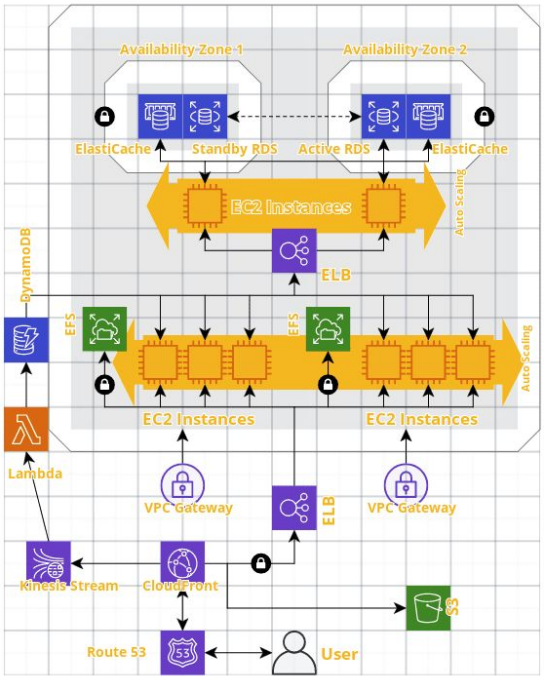
Web Application Architecture with Amazon ElastiCache and Amazon DynamoDB





Final Web Application Architecture Design





Explanations

In order to create my Final Web Application Architecture Design from scratch, I looked at five different web application architecture that are seen above, and I read through all of the benefits and drawbacks of each one. I used this information along with my previous research assessments to essentially create my final design for my Original Work, but prior to that I created these five web application architecture on CloudCraft in order to gain more experience using and designing each individual resource that I utilized. To start off, the Simple Web Application Architecture seen above is literally what the title says: simple. The Route 53 is connected to the user to translate IP Addresses to specific website information, there is only one EC2 web instance and one active RDS in one Availability Zone, and the entire thing is inside the VPC. The problem with this particular design is that it is more prone to system failures because there are not any extra components for system diversity. If one server or database fails, the entire system will fail in this design. The next Web Application Architecture with Redundancy is an updated version of the previous design, portraying more security and diversity within the VPC. There are two Availability Zones with two RDS resources in each of the zones. In case of failure in a single zone, the Multi-AZ (Multiple Availability Zones) design will ensure that the next zone continues to process. The new component seen in this design, ELB, distributes instances to certain servers and also acts like a security check that will stop the traffic if any instance fails. Altogether, this design is more elastic compared to the previous simple design, which means that system is able to handle a greater amount of load compared to the previous design. The third design shown is a Web Application Architecture with Auto Scaling. The primary purpose of the addition of Auto Scaling is to respond to the demand of the web application. If the number of

EC2 instances continue to grow in demand from the users, it is necessary to continue responding to these requests in order to have proper traffic flow. The Auto Scaling group makes it available to immediately increase or decrease traffic of EC2 instances according to the user demands of the web application. The fourth design adds Amazon CloudFront as well as Amazon S3, which is basically a storage network that can ease the load on web instances. Any JavaScript files, images, videos, style sheets, or any other heavy content from the instances can be stored in the S3. The information from the S3 can be served to the instances via the CloudFront, which is its primary function. In the fifth design, DynamoDB and ElastiCache are added to the web application to expand storage for session information in the DynamoDB and store database information in the ElastiCache. Finally, in my final design I created, I used similar resources from the examples, but I added a few extra resources. First of all, I added multiple EC2 instances in the VPC to add more connectivity and memory to the web application. Next, I added the VPC Gateways to each Availability Zone to strengthen the security of the application and prevent loss or breach of privacy in the VPC. The addition of the Kinesis Stream to my application is primarily to speed the collection and processing of large streams from the EC2 instances. Kinesis Stream is a huge factor when many instances are being monitored and there is high traffic of streaming. The Lambda was added mainly to just run code for the VPC without the need of a server. It increase serverless computing for the other resources. In the end, I added the EFS to ensure intruders stay out of the files that store data for the EC2 instances. This will protect from unauthorized access to vulnerable or sensitive data files. In conclusion, for my Original Work I created a Web Application Architecture Design that I hope to receive feedback from more professionals on and essentially add on to as the year in ISM continues to move forward.

Work Cited

Amazon Web Services (AWS) - Cloud Computing Services, https://aws.amazon.com/.

"Draw AWS Diagrams." *Cloudcraft*, https://cloudcraft.co/.

Baron, Joe. *AWS Certified Solutions Architect Official Study Guide: Associate Exam*. Sybex/John Wiley & Sons, 2017.

Kadhim, Q. K. (2018, May). A Review Study on Cloud Computing Issues. *Https://Iopscience.iop.org*. Retrieved from https://iopscience.iop.org/article/10.1088/1742-6596/1018/1/012006/pdf

Introduction to Cloud Computing. (2017). *Dialogic*. Retrieved from https://www.dialogic.com/~/media/products/docs/whitepapers/12023-cloud-computing-wp.pdf