**REPORT**

**PROJECT**

|  |  |
| --- | --- |
| **Student id** | **Name** |
| **Gaurav Dhawan** |  |
| **Manpreet singh** |  |
| **Adil sharma** |  |
| **Riya** |  |
| **Ramneet kaur** |  |
| **Prabhjot kaur** |  |

**1. Describe how this application can be deployed in a cloud environment.**

Answer:1 In order to deploy our application, we will use Admin API.

Before deploying, we, the one who is the owner of the cloud should make the app engine application but ensure that the privileges should be fulfilled by the user account. Most importantly, permission should be given to the cloud build in order to do so.

Furthermore, downloading, installing and initializing the gloud CLI is the mandatory step. In order to configure the following command should be run

gcloud.config set proxy/type [PROXY\_TYPE]

For deploying the app, following command should be run:

Gcloud app deploy [CONFIGURATION\_FILES]

**2. Describe how you would implement security features for your app in the cloud.**

Answer 2: The ways to implement the security for the app in cloud environment are as follows:

1. Managing the access to cloud applications and user behavior: Limiting the number of users as well as implementing multi-factor authentication. Moreover, using the strong policy obviously containing 14 characters such as lower case, upper case and special character. Furthermore, failed attempts of the login to the cloud should be limited.
2. Cloud governance policies: multi-factor authentication should be must with the defined roles to know who and why is the person accessing the app. Other policies include monitoring the usage etc.
3. Malware threat protection: The advanced components used by the attackers make this protection quite difficult. In order to protect, create a BYOD protection policy which helps to secure upload. Even, upgrade a cloud-specific protective layer to the cloud application so that the infrastructure is secured.

**3.)If you need to make your application serverless how it can be done.**

Answer 3: In order to build the serverless application, we can use AWS Lambda, Amazon API Gateway, Amazon DynamoDB, Amazon Cognito.

This can be done in the following order:

Firstly, hosting a static website to configure the AWS Amplify. Eventually, managing the users by creating amazon cognito user pool is mandatory step. Furthermore, building a serverless backend is utter most important to handle requests. Then, use amazon API gateway. Finally, terminate the resources.

REFERENCES:

<https://cloud.google.com/appengine/docs/legacy/standard/python/tools/uploadinganapp>

<https://www.cypressdatadefense.com/blog/application-security-in-cloud/>

https://aws.amazon.com/getting-started/hands-on/build-serverless-web-app-lambda-apigateway-s3-dynamodb-cognito/