

|  |  |  |
| --- | --- | --- |
| **Page Info:** |  |  |
| **Page Name** | **Deploy a flask app on Amazon AWS EC2 and keep it running while you are offline.** | |
| **Content Creator** |  | |
| **Approver** |  | |
| **Word Count** |  | |
| **Live URL** | https://medium.com/analytics-vidhya/deploy-a-flask-app-on-amazon-aws-ec2-and-keep-it-running-while-you-are-offline-38d22571e2c5 | |
| **New URL** | **/** | |
| **SEO Data:** |  |  |
| **Page Title (40-60 Characters )** | Deploy a flask app on Amazon AWS EC2 and keep it running while you are offline. | by Ankur Bhatia | Analytics Vidhya | Medium | |
| **Meta Description (140-160 Characters)** | Flask: Flask is a micro web framework in python. It is designed to make getting started quick and easy, with the ability to scale up to complex applications. It’s easy to get started with to deploy… | |
| **Focus Keyword** |  | |
| **Key Variations** |  | |
| **Sprinkle Keywords (Use if you can, not required)** |  | |
| Misc/Notes |  | |
| Authority Score | 89 | |
| Backlinks | 889188527 | |
| **Onpage:** |  |  |
| Content | Analytics Vidhya  Apr 15, 2020  Flask:Flask is a micro web framework in python. It is designed to make getting started quick and easy, with the ability to scale up to complex applications. It’s easy to get started with to deploy .py’s on the web.  Amazon Web Services EC2:You can google more on it. In simple terms — it provides you access to a computer, kept far away, running and connected all time with internet. You can choose the machine from different specifications available depending upon the compute you require. Basic machines are available for free, complex specifications are at a cost.  Lets Get Started -  Step 1: Create an account on AWS For getting started, you’ll require a credit card. But you won’t be charged for free instances (which we will be using).  Step 2: Navigate to AWS Management Console, Services and click on EC2 in Compute Services.  Step 3: Navigate to Instances on the side bar and click on Launch Instance.  Choose the required machine according to requirements. I am choosing the Ubuntu 18.04 LTS Server.  Step 4: Choose the TF2.micro type instance and click on ‘Review and Launch button’ at the bottom right.  Step 5: You can review your system configurations here. Step up and click on Launch at the bottom right.  Step 6: Select the ‘Create a new key pair’ option and provide with the ‘Key pair name’ of your choice. (1) Download the Key Pair (don’t lose it, it’ll be useful in accessing the machine).(2) After downloading, click on ‘Launch Instances’.(3) Click on ‘View Instances’ on the next page.  Step 7: Wait until the instance gets initialized.  You can give a name to your instance as per your choice.  Step 8: Copy the IPv4 Public IP address.  Step 9: Now we need to ssh in this instance from our local pc.1. To do this, first we need to open the terminal and navigate to the directory where we have the downloaded key pair file (.pem file)2. Now give read only permissions to the key pair file using the following command: chmod 400 <your-pem-file-name>.pem(it’ll provide read-only permissions to the file)3. And using the copied public key, SSH into the AWS instance using this command: ssh -i <your-pem-file-name>.pem <your-machine-type>@<your-public-ip>  Step 10: Update the system: sudo apt-get updateCheck for python3: python3 -V (if not present -then install)Check for pip3/ install using: sudo apt install python3-pipInstall Flask using: pip3 install flaskAlso install other dependencies of your flask app.  Step 11: After Installation of the dependencies, Create a file named server.py and add the contents of your flask app. Here I am using just a simple app with the following contents. \*Check the host and the port no.  from flask import Flaskapp = Flask(\_\_name\_\_)  @app.route('/')def hello\_world(): return 'Hello, World!'  if \_\_name\_\_ == "\_\_main\_\_": app.run(host="0.0.0.0", port=80)  Step 12: For your Flask application Instance, click on the ‘launch wizard’ in the ‘Security groups’ options in the Description tab.  Step 13: Click on ‘Inbound Rules’ → ‘Edit Inbound Rules’Add a new rule (using ‘Add rule’ button) with Type: HTTP, Port: 80, host: 0.0.0.0/0 and ::/0  Step 14: This is the important step for running of the Flask Application.From the terminal, you can run the flask application using the command:sudo python3 server.py(Make sure you are ssh’ed to the Instance)But this command will get automatically killed if we closed the terminal, or exited from the ssh to the instance.To keep running the application (so that you may close your laptop and have some fun, while the application keeps running), we will use the powerful linux command: nohup (no hangup).So for running the python application we will use the command: nohup python3 server.py &(& allows us to run the application in background and nohup allows the application to keep running even on hang up/logout)  We can see after the nohup command the ‘process id’ : 21108 of the running process. We can use the command pgrep <process-type> to check the processes running of the ‘process-type’. Now we can logout from the terminal/system, but our process will be running. To check the process running, paste the IPv4 public IP address of the instance in the browser. In Our case its: 35.154.90.20  Here we are! Live!  Now if we wish to kill the process we can use the command: sudo kill <process-id>. So to kill this process we can use the command: sudo kill 21108. (Make sure you are ssh’ed in the instance)  Please leave suggestions and feedback in the comments. If you have any query, drop me a mail at: bhatia.ankur24@gmail.com  --  --  More from Analytics Vidhya  Analytics Vidhya is a community of Analytics and Data Science professionals. We are building the next-gen data science ecosystem https://www.analyticsvidhya.com  Recommended from Medium  Aditya Pande  Setting up Python Interpreter and running Python Code on Docker Container  Asger Arnbjørn Thyregod  in  AdNudging  How We Integrated Google Prosperworks, Calendly and Mailchimp  Norman Benbrahim  Turning News Into Data — Day 7  Lane Wagner  in  Boot.dev  Learn Go Fast; Best Courses and Resources  Dominik Spieler  My journey of starting programming - making money with bots.  Evgeny Grishchenko  in  Dev Genius  DAPR — Part 2  Vanchy  My CKAD Journey  Mercy Ben  How to build a Slack bot with Python  AboutHelpTermsPrivacy  Get the Medium app  Ankur Bhatia  Working in the field of Deep Learning and Robotics Development.  Help  Status  Writers  Blog  Careers  Privacy  Terms  About  Knowable | |