Graph.Build Studio Usage Instructions

Usage instructions

These usage instructions will help ensure the successful configuration and running of the Graph.Build Studio.

Using the 1-Click Quick Create Stack setup

Using a cloudformation template, which can be found here, we can utilise a Quick Create Stack to launch the Graph.Build Studio with 1-Click.

1. Launch the product via 1-Click Quick Create Stack
   - Small: https://us-east-1.console.aws.amazon.com/cloudformation/home?region=us-east-1#/stacks/create/review?templateURL=https://data-lens-cloudformation.s3.amazonaws.com/Studio-Templates/studio-public-s.yaml&stackName=GraphBuildStudio-Small
   - Medium: https://us-east-1.console.aws.amazon.com/cloudformation/home?region=us-east-1#/stacks/create/review?templateURL=https://data-lens-cloudformation.s3.amazonaws.com/Studio-Templates/studio-public-m.yaml&stackName=GraphBuildStudio-Medium
   - Large: https://us-east-1.console.aws.amazon.com/cloudformation/home?region=us-east-1#/stacks/create/review?templateURL=https://data-lens-cloudformation.s3.amazonaws.com/Studio-Templates/studio-public-l.yaml&stackName=GraphBuildStudio-Large
   - More information on the Cloudformation template can be found below
2. Once deployed, use a web browser to access the application at `http://<EC2_Instance_Public_DNS>`, this link can be found in the resources tab on your AWS console.

3. Sign in using the following credentials:
   - Username: `admin`
   - Password: the default password provided in the quick create
     i. Note: When signed in, it is best practice to change this password in the Admin Panel user setting

Manually setting up your stack

1. Create an ECS Cluster with an EC2 instance of the desired size (usually t2.medium or t2.large) with an EBS volume of 30GB attached (that is the minimum EBS volume size allowed with ECS and should be more than enough for storing the Studio data)
2. Place the instance into a public subnet in a VPC with an internet gateway
3. Create a security group to allow access to the EC2 instance from your personal ip
4. Create a task definition to run the UI and UI backend as containers using the ec2 instances public ip as part of the `REACT_APP_NODEAPP_HOST` environment variable to allow the react UI to communicate with the Node backend, e.g.

```
{
      "name": "REACT_APP_NODEAPP_HOST",
      "value": "http://<your-ec2-public-ip>:9601"
}
```

Note that this is the most basic initial setup. For improved security we recommend creating a service with an application loadbalancer attached that uses your cluster. The load balancer should have a SSL/TLS certificate attached to it. Amazon Cognito can be integrated with the Load balancer to authenticate user requests against an AWS user pool. The task can then be securely run by the service.

Upgrade instructions

1. All data processed by the Studio is stored in the `/dist/database/sqlite/gbstudio.db` database file. To preserve your data during upgrade, please ensure this volume is not deleted.
   - Note: when deleting a stack created by the 1-Click deployment, the data volume will not be deleted
2. When setting up your new instance, ensure your volume is the same as your previous one.
3. It is recommended to make a backup of your data by downloading a copy of your `gbstudio.db` file.

CloudFormation Delivery Information

- All sensitive information is saved within your environment and nothing is shared with Graph.Build
- IAM roles and Policies: The template includes roles for allowing the ECS tasks to access metered usage, upload and down from S3, Download images from ECR, upload logs to Cloudwatch, Create ECS Clusters, Services and Tasks, Create and write to an EBS volume.
- Network Configuration: The template creates a VPC with 2 public subnets. A loadbalancer sits across the 2 subnets and forwards requests to a single EC2 instance that is running the UI and UI backend as part of an ECS cluster. Requests are routed by port to the relevant container.
- AWS resources pricing breakdown:

Resource	**$ hourly**	**$ month**
t2.medium	0.0464	33.87
EBS Volume 30GB	0.0032	2.4

Detailed monitoring	0.0032	2.4
Application LoadBalancer	0.0225	16.43

Monitoring and assessing application functions

1. Navigate to your Amazon EC2 console and verify that you're in the correct region.
2. Choose Instances and select your launched instance.
3. Select the server to display your metadata page and choose the Status checks tab at the bottom of the page to review if your status checks passed or failed.

User Documentation

For further instructions please contact support@graph.build to get access to our user documentation at https://docs.graph.build.