RoostGPT Infrastructure

This section details the RoostGPT resources required

- RoostGPT Resources for Whitelisting
- RoostGPT Database for Hosted Server

RoostGPT Resources for Whitelisting

Terraform Resources Provisioned in AWS

Resource Name	Description
aws_key_pair	AWS EC2 keypair to SSH
aws_eip	An elastic ip to be used by NAT Gateway for public subnet
aws_ami	Ubuntu based AMI id - region specific
aws_vpc	VPC
aws_availability_zones	AZ preferred
aws_internet_gateway	IG for VPC to communicate with internet e.g. apt update
aws_subnet	private and public subnets
aws_security_group	SG for bastion, backend and frontend service of RoostGPT
aws_lb_target_group	Target group for loadbalancer
aws_route_table	Route Table entries for inbound and outbound traffic
aws_instance	Instances to deploy RoostGPT frontend and backend services
aws_route_table_association	To allow network traffic for instance to instance communication
null_resource	To deploy and run RoostGPT installer scripts on instance

Docker Images (hub.docker.com)

Image Name	Tag
zbio/roostai_mysql_db	v1.1.17
zbio/roost-nginx	latest
zbio/roost-app	v1.1.17
zbio/roostai-server	v1.1.17
zbio/roostgpt-go	v1.1.17
zbio/roostgpt	v1.1.17
zbio/roostgpt-ui	v1.1.17
zbio/roost-proxy	v1.1.17

RoostGPT scripts or executables from

https://github.com/roost-io/roost-support/releases/tag/v1.1.17

Name	Purpose
roost.sh	Deployment script for Hosted stack
roost-enterprise.sh	Controller script for Hosted stack
main-config.json	Configuration Template for Hosted stack
aiServer.sh	Internal scripts for Hosted stack
aiServer.gz	Internal daemon executable for Hosted stack
releaseServer.sh	Internal scripts for Hosted stack
releaseServer.gz	Internal daemon executable for Hosted stack
roost.sql	SQL template for Database (Hosted stack)
roostcertgen.gz	Self signed certificate generation (Hosted stack)
roost-deployment-tool	Installer for On-Prem, single Ubuntu server hosted stack
roostgpt-1.0.15.vsix	VS Code Plugin

	Windows	Linux	MacOS
CLI	roostgpt-win.exe	roostgpt-linux	roostgpt-macos
UI Test	RoostUITestGenerator-win.exe	RoostUITestGenerator-linux	RoostUITestGenerator-macos-amd RoostUITestGenerator-macos-arm
Java	RoostJavaASTParser.jar	RoostJavaASTParser.jar	RoostJavaASTParser.jar
Python	RoostPythonASTParser-win.exe	RoostPythonASTParser-linux	RoostPythonASTParser-mac
CSharp	RoostCSharpASTParser-win.exe	RoostCSharpASTParser-linux	RoostCSharpASTParser-macos
Golang	RoostGolangASTParser-win.exe	RoostGolangASTParser-linux	RoostGolangASTParser-macos
Javascript	RoostJavascriptASTParser-win.exe	RoostJavascriptASTParser-linux	RoostJavascriptASTParser-macos
API (Karate Test)	karate-1.4.1.jar karate-1.5.1.jar	karate-1.4.1.jar karate-1.5.1.jar	karate-1.4.1.jar karate-1.5.1.jar

RoostGPT Database for Hosted Server

RoostGPT solution requires a RDBMS to

- Store test configurations
- RoostGPT connector information
- User and Team information

For POC or limited trials, RoostGPT can also use a containerised database. However, it has limitations related to

- Enterprise-grade reliability and security
- Advanced feature set and integrations
- Professional support and SLA guarantees
- Automated maintenance and updates