



Docker and DevOps

Who am I?

- My name is Samiul Haque, I am a developer working at RBC (Royal Bank of Canada) in the Developer Experience team.
- When I am not hacking away and starting fires at work, I make cool projects such as slackbots and privatized voice applications to help aid our day-day operations and user experience.
- Outside of work, I am known for creating high voltage linear accelerators, photography and general tech support for my family and friends.

What are we doing today?

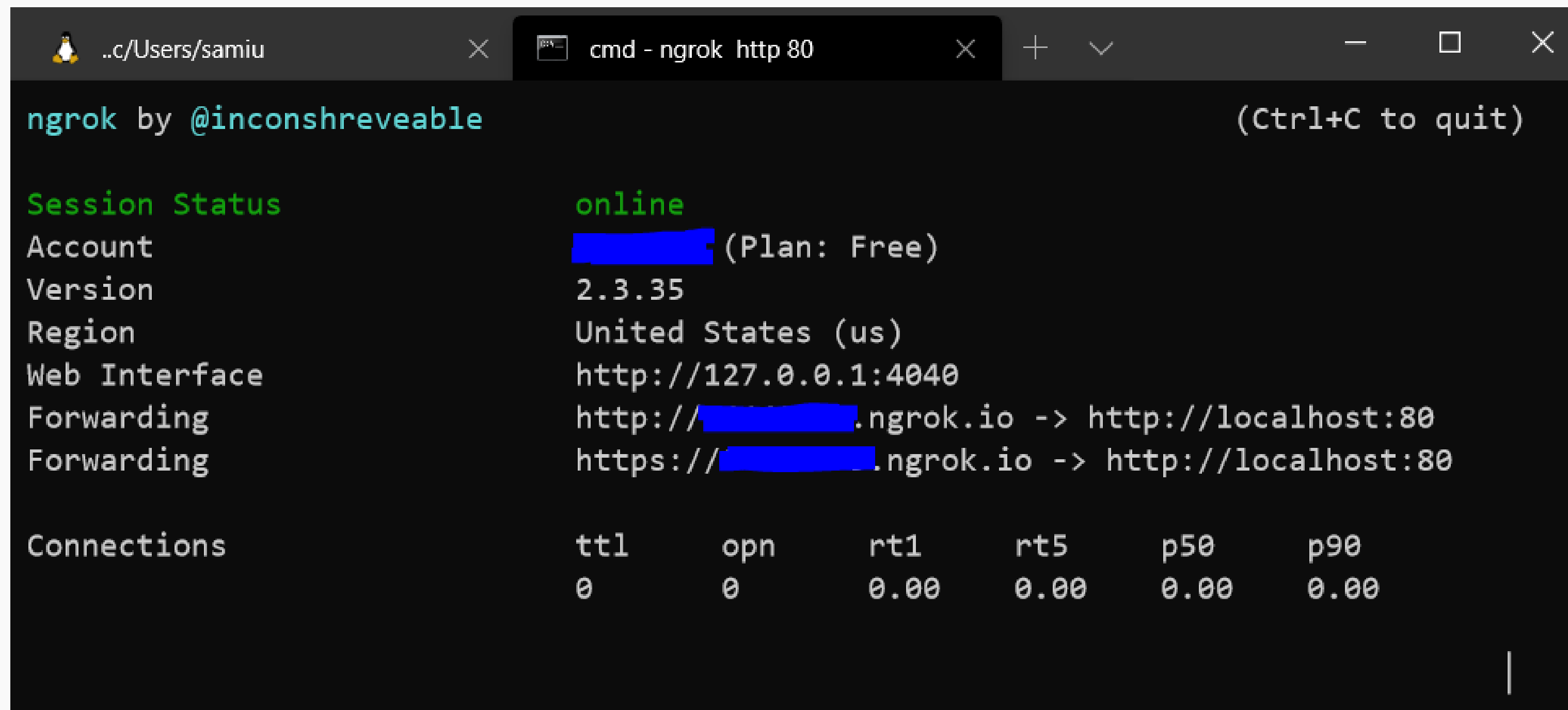
- Today we will be building a simple python flask web application which receives JSON formatted triggers from Slack when a '/' slash command is triggered and outputs a GIF of a gnome.
- To allow easier testing of the application on localhost we will be using ngrok.
- I chose a gnome as my friend who is also attending this hackathon hates gnomes, and I routinely taunt him by sending a gnome gif for fun XD.
- We will then be pushing our code to Github and then setting up our CI/CD pipeline using Github actions and publishing our docker image to docker hub.
- Lastly, we will be deploying our final image on a DigitalOcean VM.

Setting up ngrok and Slack bot

- We need ngrok to give us a public url so we can test our application to see if it gives the correct response back to Slack
- Go to: <https://api.slack.com> to first create an application leveraging a slash command. Detailed instructions can be found here: <https://api.slack.com/interactivity/slash-commands>
- Next, we need to download ngrok to proxy requests from a public url that slack can send to and proxy it into our localhost. Detailed instructions on how to setup ngrok can be found here: <https://dashboard.ngrok.com/get-started>

Ngrok output

- Once you have ngrok setup you should see a connected dialog by running the command: `ngrok http 5000`

A screenshot of a terminal window with a dark background. The window title bar shows two tabs: the first is labeled with a penguin icon and the path '..c/Users/samiu', and the second is labeled 'cmd - ngrok http 80'. The terminal content shows the ngrok logo and version, followed by session status information including account details, version, region, web interface URL, and forwarding rules. At the bottom, there is a table showing connection statistics.

```
ngrok by @inconshreveable (Ctrl+C to quit)

Session Status      online
Account             [REDACTED] (Plan: Free)
Version             2.3.35
Region              United States (us)
Web Interface       http://127.0.0.1:4040
Forwarding           http://[REDACTED].ngrok.io -> http://localhost:80
Forwarding           https://[REDACTED].ngrok.io -> http://localhost:80

Connections          ttl      opn      rt1      rt5      p50      p90
                    0         0        0.00     0.00     0.00     0.00
```

Slack output

- Once you have the slack application setup and the bot scope setup you should paste the ngrok forwarding url from the previous step to the slash commands target url. As shown below:

Edit Command [X]

Command: /gnome ⓘ

Request URL: https://[redacted].ngrok.io ⓘ

Short Description: gnomes channel

Usage Hint: [which rocket to launch]
Optionally list any parameters that can be passed.

Escape channels, users, and links sent to your app ☒

Escaped: <@U1234|user> <#C1234|general>

Preview of Autocomplete Entry

Commands matching "gnome"

GnomeBot

Cancel Save

The code

<https://github.com/samhaque/GnomeBot>

```
import os

from flask import Flask, request, jsonify

VERIFICATION_SECRET = os.getenv("VERIFICATION_TOKEN")

app = Flask(__name__)

@app.route('/', methods=["POST"])
def gnome_channel():
    print(request.form)
    if request.form['token'] == VERIFICATION_SECRET:
        return jsonify({
            "response_type": "in_channel",
            "attachments": [
                {
                    "title": "You've been Gnomed",
                    "image_url": "https://thumbs.gfycat.com/SaneDisgustingAtlanticridleyturtle-max-1mb.gif"
                }
            ]
        })
    return jsonify({
        "response_type": "ephemeral",
        "text": "Sorry, that didn't work. Please try again."
    })

if __name__ == '__main__':
    app.run(host="0.0.0.0", debug=False)
```

Helpful Links

- <https://csec.club/resources>
- <https://github.com/features/actions>
- <https://docs.docker.com/get-started/>
- <https://kubernetes.io/docs/tutorials/>
- <https://www.freecodecamp.org/news/a-beginner-friendly-introduction-to-containers-vms-and-docker-79a9e3e119b/>
- <https://www.atlassian.com/devops>
- <https://blog.sqreen.com/secdevops/>

Live deploy on Digitalocean VM

.

Q&A