Developing and Deploying Intelligent Chat Bots

Microsoft

Ngrok

Objectives

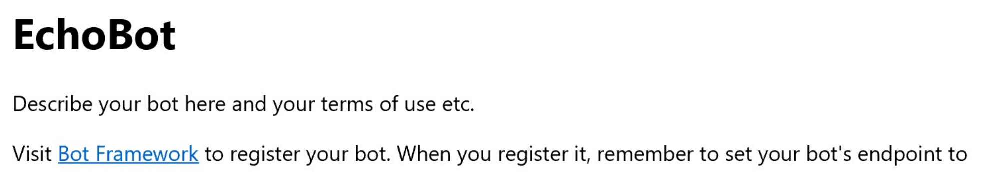
With Microsoft Bot Framework, to configure the bot to be available to a particular channel, you will need to host the Bot service on a public URL endpoint. The channel won't be able to access your bot service if it is on a local server port hidden behind a NAT or firewall.

When designing / building / testing your code you don't always want to have to keep redeploying. This will result in additional hosting costs. This is where ngrok can really help in speeding up the development/testing phases of bots.

The goal of this lab is to use ngrok to expose your bot to public internet and use the public endpoints to test your bots in the emulator.

Setup

1. If you do not have ngrok installed, download ngrok from https://ngrok.com/download and install for your OS
2. Open any Bot project (from the previous labs) in Visual Studio and run it. For the EchoBot, you should be seeing the below message in the browser:



EchoBot

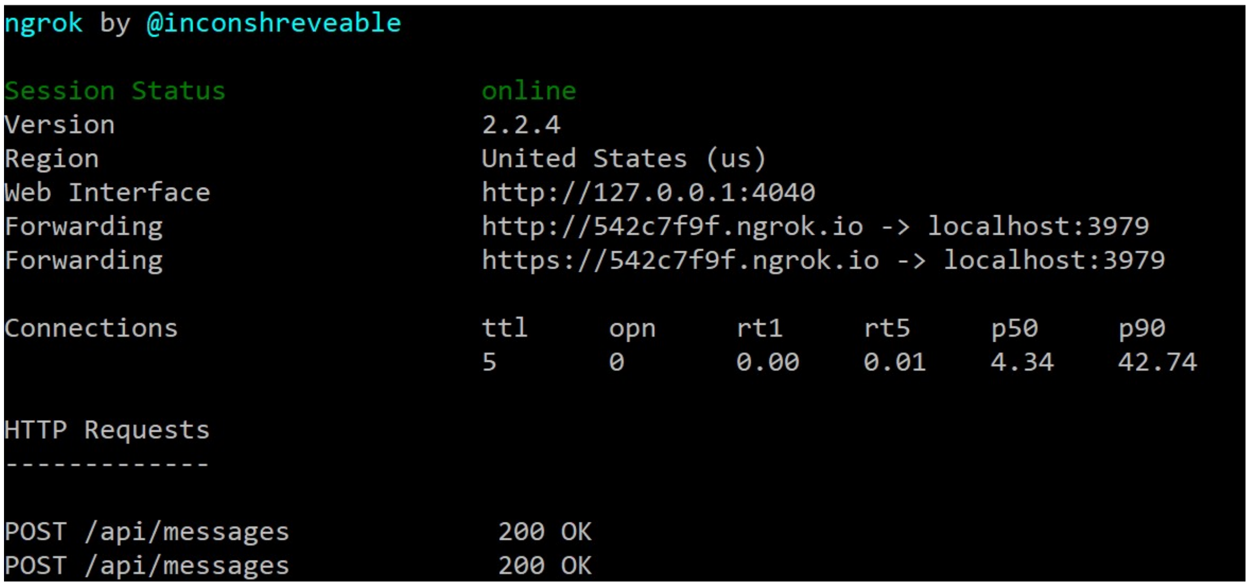
Describe your bot here and your terms of use etc.

Visit Bot Framework to register your bot. When you register it, remember to set your bot's endpoint to

Forwarding

1. Given the bot is being hosted on localhost:3979, we can use ngrok to expose this port to the public internet
2. Open terminal and go to the folder where ngrok is installed
3. Run the below command and you should see the forwarding url:

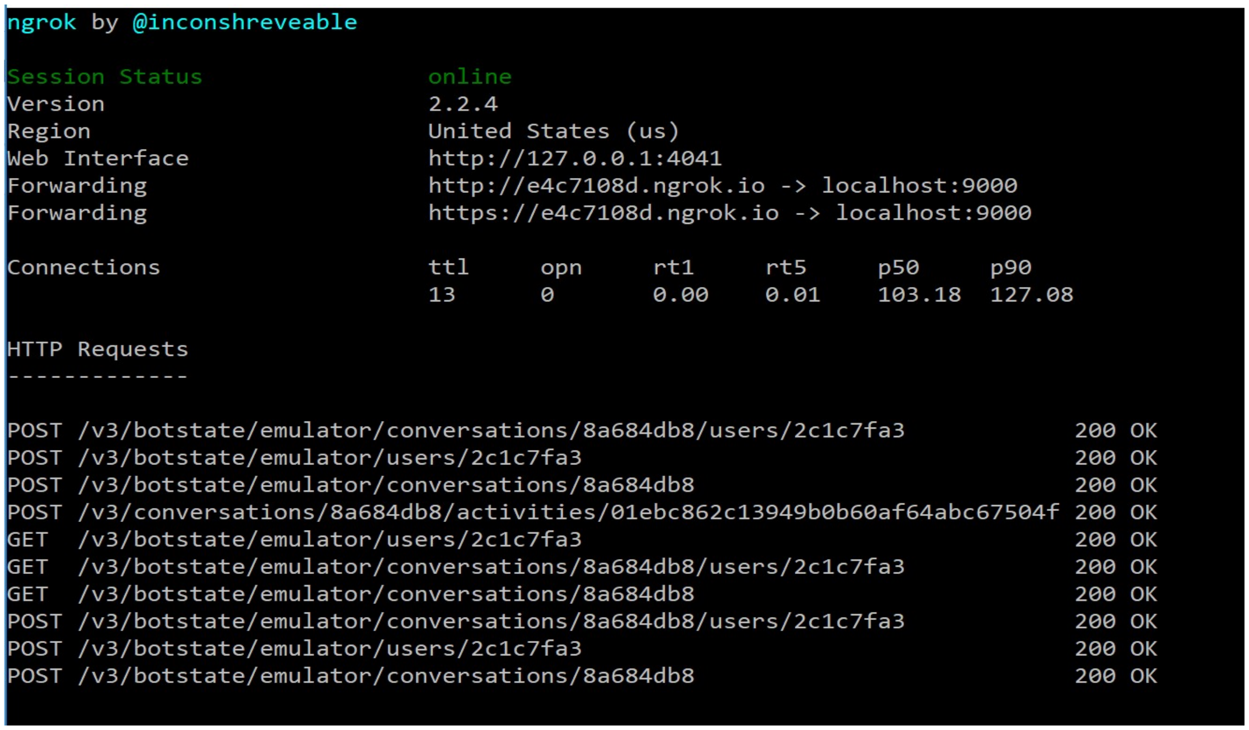
```
ngrok.exe http 3979 -host-header="localhost:3979"
```



```
ngrok by @inconshreveable

Session Status                online
Version                       2.2.4
Region                        United States (us)
Web Interface                 http://127.0.0.1:4040
Forwarding                    http://542c7f9f.ngrok.io -> localhost:3979
Forwarding                    https://542c7f9f.ngrok.io -> localhost:3979

Connections                   ttl     opn     rt1     rt5     p50     p90
                              5       0       0.00    0.01    4.34    42.74

HTTP Requests
-------------

POST /api/messages             200 OK
POST /api/messages             200 OK
```

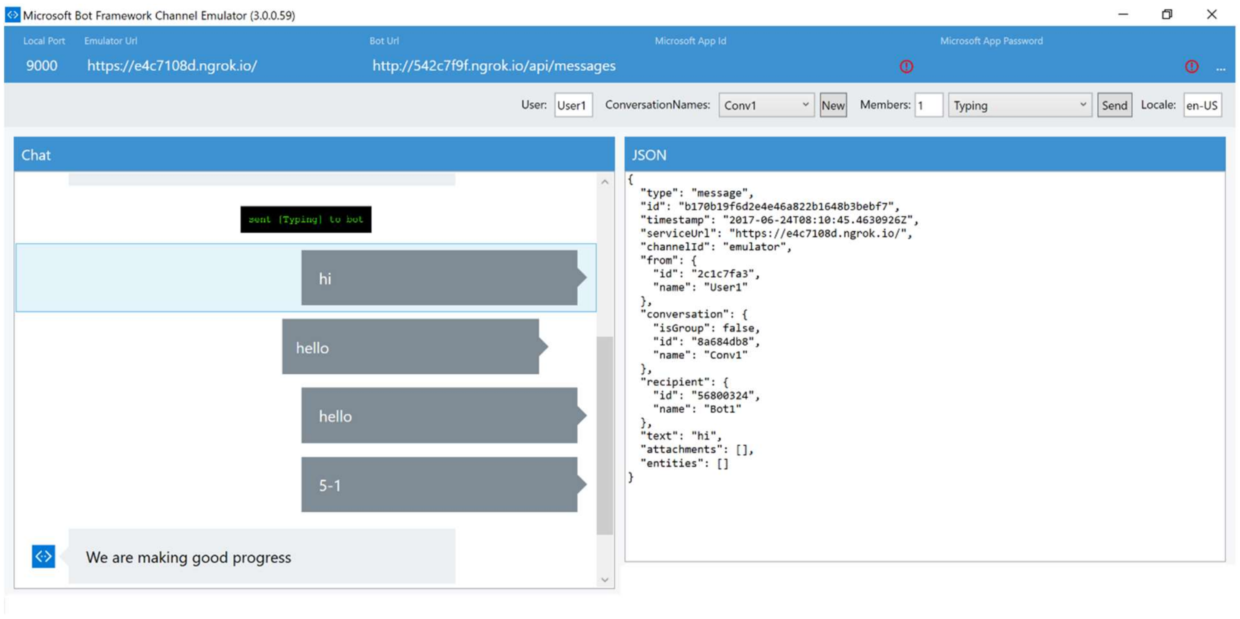
4. To use public urls in the bot emulator, you will also need to generate a forwarding url using ngrok for Emulator url (port 9000). Run the below command and you should see the forwarding url for port 9000:

   ngrok.exe http -host-header=rewrite 9000



```
ngrok by @inconshreveable

Session Status                online
Version                       2.2.4
Region                        United States (us)
Web Interface                 http://127.0.0.1:4041
Forwarding                    http://e4c7108d.ngrok.io -> localhost:9000
Forwarding                    https://e4c7108d.ngrok.io -> localhost:9000

Connections                   ttl     opn     rt1     rt5     p50     p90
                              13      0       0.00    0.01    103.18  127.08

HTTP Requests
-------------

POST /v3/botstate/emulator/conversations/8a684db8/users/2c1c7fa3            200 OK
POST /v3/botstate/emulator/users/2c1c7fa3                                   200 OK
POST /v3/botstate/emulator/conversations/8a684db8                           200 OK
POST /v3/conversations/8a684db8/activities/01ebc862c13949b0b60af64abc67504f 200 OK
GET  /v3/botstate/emulator/users/2c1c7fa3                                   200 OK
GET  /v3/botstate/emulator/conversations/8a684db8/users/2c1c7fa3            200 OK
GET  /v3/botstate/emulator/conversations/8a684db8                           200 OK
POST /v3/botstate/emulator/conversations/8a684db8/users/2c1c7fa3            200 OK
POST /v3/botstate/emulator/users/2c1c7fa3                                   200 OK
POST /v3/botstate/emulator/conversations/8a684db8                           200 OK
```

5. Enter the forwarded urls in the bot emulator (bot url and emulator url). The bot url will have /api/messages appended to the forwarding url. Test the bot in the emulator by sending messages.



hi

hello

hello

5-1

We are making good progress

```
{
  "type": "message",
  "id": "b170b19f6d2e4e46a822b1648b3bebf7",
  "timestamp": "2017-06-24T08:10:45.4630926Z",
  "serviceUrl": "https://e4c7108d.ngrok.io/",
  "channelId": "emulator",
  "from": {
    "id": "2c1c7fa3",
    "name": "User1"
  },
  "conversation": {
    "isGroup": false,
    "id": "8a684db8",
    "name": "Conv1"
  },
  "recipient": {
    "id": "56800324",
    "name": "Bot1"
  },
  "text": "hi",
  "attachments": [],
  "entities": []
}
```

Exercise

1. When you register the bot on the Microsoft Bot Framework, can you use the forwarding url for the Messaging Endpoint?
2. Test the bot on a channel using the forwarding URL on a channel.