To access two remote systems using **Ngrok**, you can establish tunnels to each machine separately, as Ngrok's free version only supports a single tunnel at a time. For two systems that don't have public IPs and are on different local networks, you will need to:

1. Run **Ngrok** on each machine to create separate tunnels.
2. Use the Ngrok tunnels to expose the services running on each machine to the internet.

**Step-by-step outline:**

1. **Install Ngrok on both machines**: Download and install Ngrok on each machine from here.

Once downloaded, you can install Ngrok by unzipping the downloaded file and moving it to a directory in your $PATH.

1. **Start a service on both machines** (for example, a web server or any application that you want to expose to the internet).

Let's say you want to run a simple Python HTTP server on each machine:

bash

Copy code

python3 -m http.server 8000

This command will run a basic HTTP server on port 8000.

1. **Expose each machine's local service via Ngrok**.

On **Machine 1**:

bash

Copy code

ngrok http 8000

This will create an HTTP tunnel and give you a public URL, like http://abcd1234.ngrok.io, that forwards traffic to your local service on port 8000.

On **Machine 2**: Similarly, you can run Ngrok to expose the local service on port 8000 on **Machine 2**:

bash

Copy code

ngrok http 8000

This will create a different tunnel, with a URL like http://efgh5678.ngrok.io.

1. **Access both machines from anywhere**. Now you can access the services running on both machines using their respective Ngrok URLs:
   * **Machine 1**: http://abcd1234.ngrok.io
   * **Machine 2**: http://efgh5678.ngrok.io

These URLs will allow you to access the services on these machines over the internet, even though they are not on the same local network and do not have public IPs.

**Python script**

remoteaccess\_ngrok.py