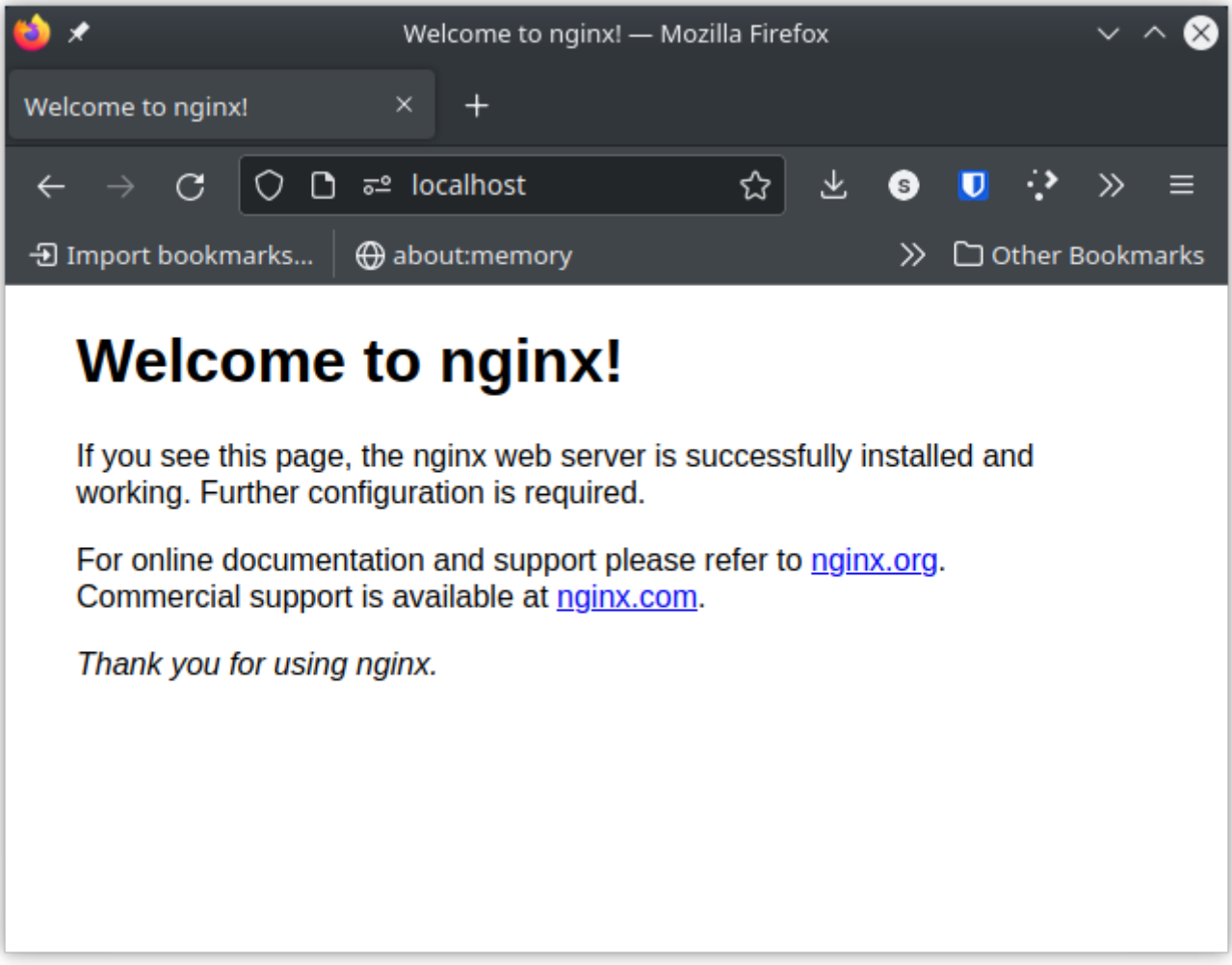
Tunneling

Reverse Tunneling

this is AKA Remote Port Forwarding, where we basically forward a remote server's port to direct requests to a local port on our machine. This is rather fun to play with, and only takes a few minutes to complete a working example if you're familiar with Linux and NGINX.

First we should keep in mind that if we want to forward any ports below `1024` on the remote server, we need to login as the root user. It doesn't matter if your user has sudo or not, it won't work unless you are root. You could maybe reconfigure things to make this not the case, but for the sake of this example we will just use the root user.

Start a local NGINX server and visit `localhost` in your web browser to see that it's working correctly. We will just use the default NGINX template.



Now login to your remote server and make sure the following line is withi `/etc/sshd/sshd_config` to allow public port forwarding.

```
# /etc/sshd/sshd_config
# By default, this is set to `no`; Make sure you change it to `yes`
# GatewayPorts no
GatewayPorts yes
```

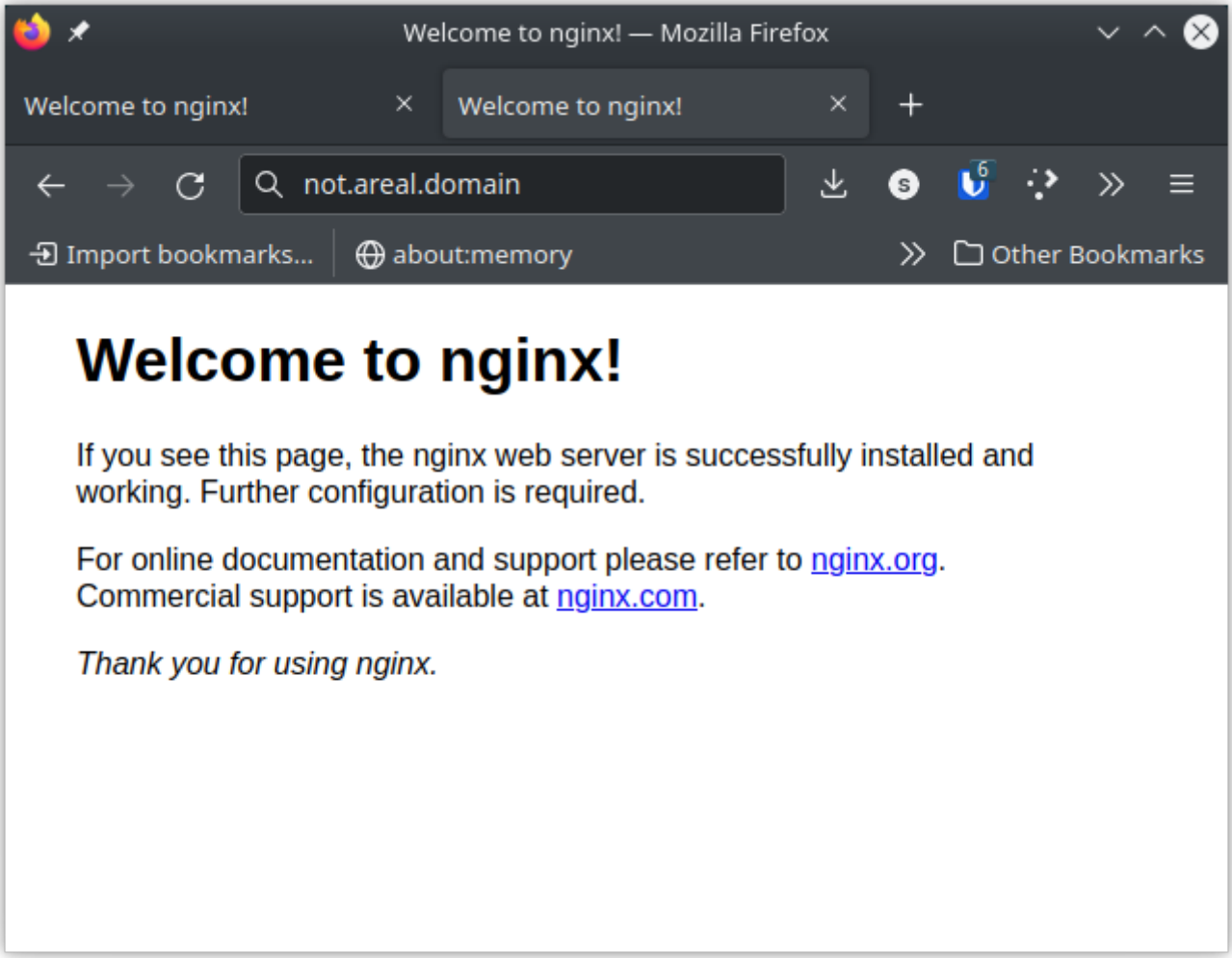
Now, restart the `sshd.service` by running the following command, and make sure to stop the `nginx.service` if it is running on your remote server. Finally we `exit` the ssh session so we can relog as `root` and start our remote SSH tunnel

```
sudo systemctl restart sshd.service
sudo systemctl stop nginx.service
exit
```

To bind the remote server with the ssh command, the syntax is `ssh -R <REMOTE_PORT>:<LOCAL_IP>:<LOCAL_PORT> root@<REMOTE_IP>`. An example of this for my server is the command below. Note the remote IP is fake, since I don't want to share this IP publicly.

```
ssh -R 80:127.0.0.1:80 root@123.456.789.123
```

That's it! Once you've connected to your ssh session, you can visit your remote server's domain name or IP and it will redirect requests to port `80` to your local webserver.



Sources

goteleport - ssh tunneling explained

Revision #1
Created 21 April 2022 12:50:42 by Shaun Reed
Updated 21 April 2022 13:04:38 by Shaun Reed