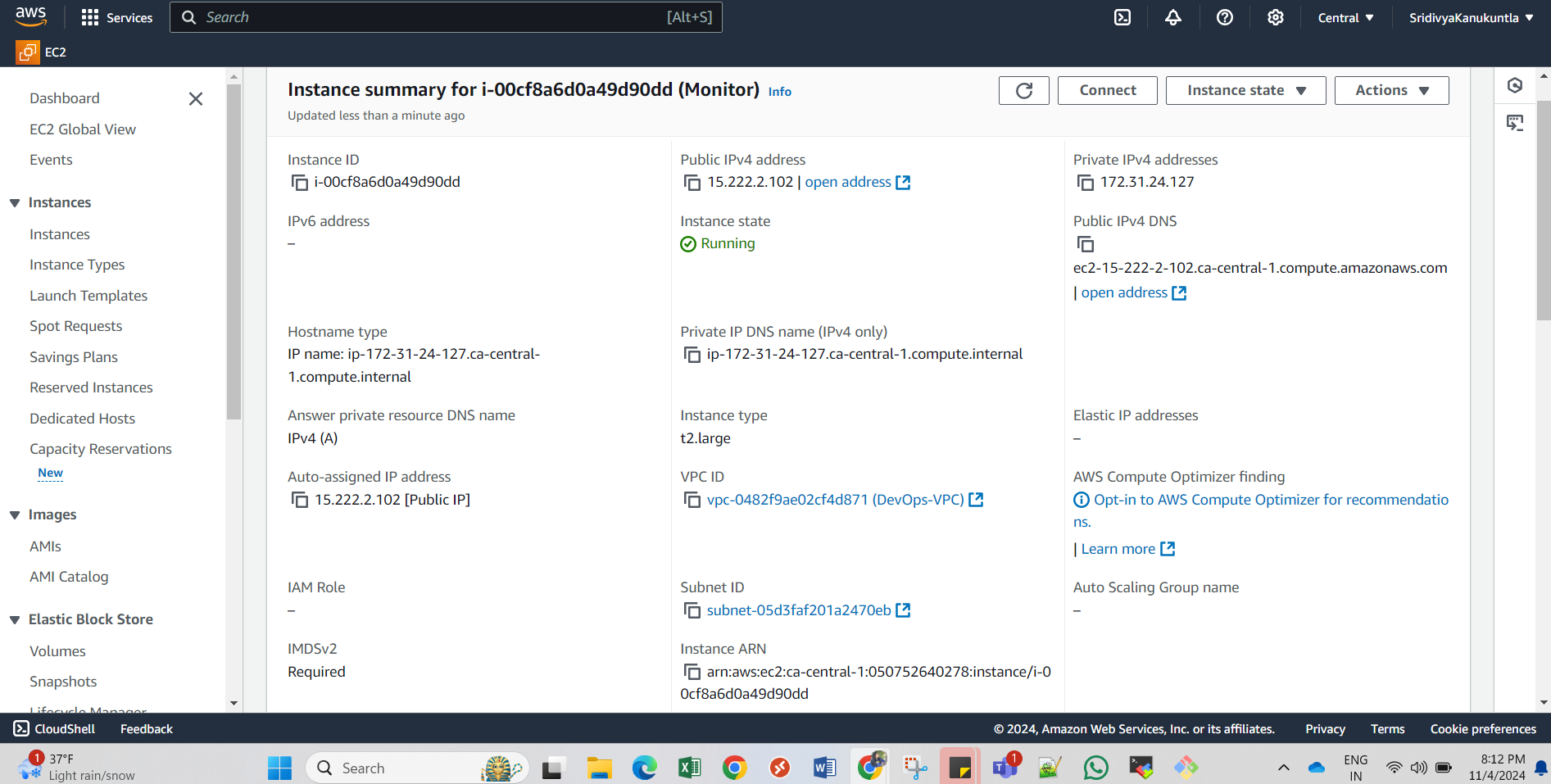
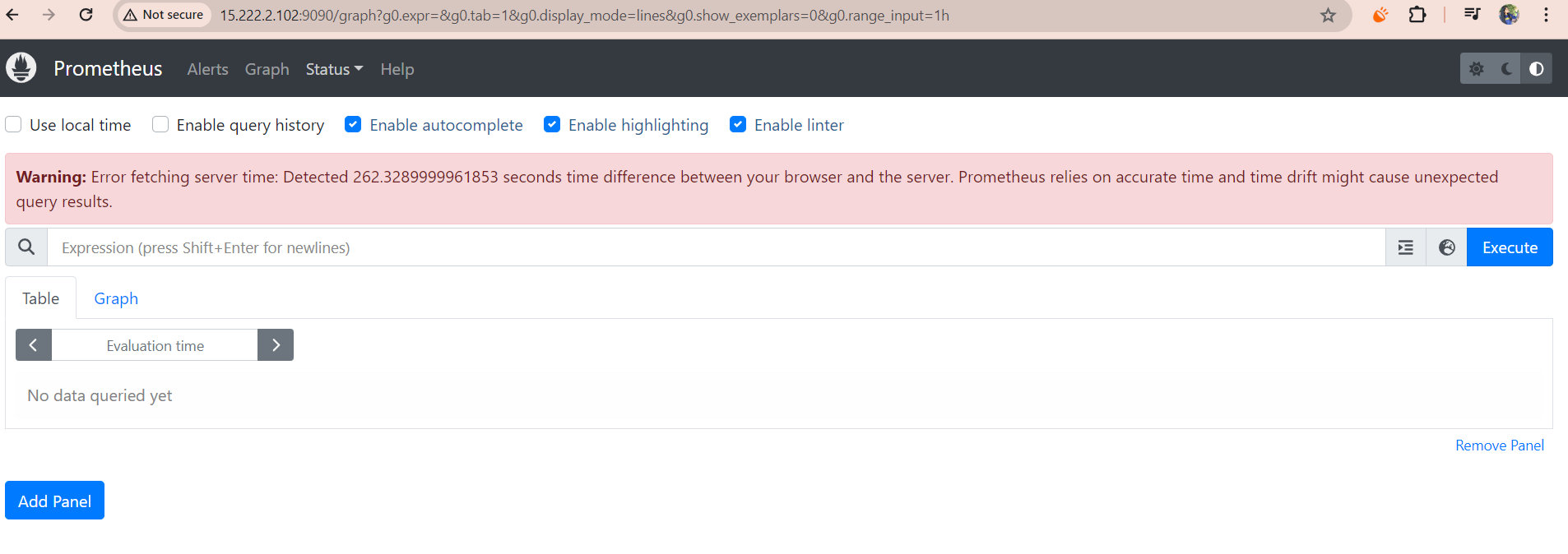
Phase4: Setting up Monitoring

***Step1: Create an Instance for Monitoring purpose as shown below:***

***Step2: Install Prometheus***

* Update the package list:
  + sudo apt update
* Downlaod Prometheus:
  + wget <https://github.com/prometheus/prometheus/releases/download/v2.55.0/prometheus-2.55.0.linux-amd64.tar.gz>
* Extract the downloaded file:
  + tar -xvzf prometheus-2.55.0.linux-amd64.tar.gz
* Start Prometheus
  + ./Prometheus &
* Open Prometheus UI in web browser using prom server IP on default port 9090
  + http://<prometheus-server-ip>:9090



***Step3: Install Grafana***

* Run the below commands on the monitoring server:
  + sudo apt-get install -y adduser libfontconfig1 musl  
    wget <https://dl.grafana.com/enterprise/release/grafana-enterprise_11.3.0_amd64.deb>  
    sudo dpkg -i grafana-enterprise\_11.3.0\_amd64.deb

O/P:

ubuntu@ip-172-31-24-127:~$ sudo dpkg -i grafana-enterprise\_11.3.0\_amd64.deb

Selecting previously unselected package grafana-enterprise.

(Reading database ... 67987 files and directories currently installed.)

Preparing to unpack grafana-enterprise\_11.3.0\_amd64.deb ...

Unpacking grafana-enterprise (11.3.0) ...

Setting up grafana-enterprise (11.3.0) ...

info: Selecting UID from range 100 to 999 ...

info: Adding system user `grafana' (UID 111) ...

info: Adding new user `grafana' (UID 111) with group `grafana' ...

info: Not creating home directory `/usr/share/grafana'.

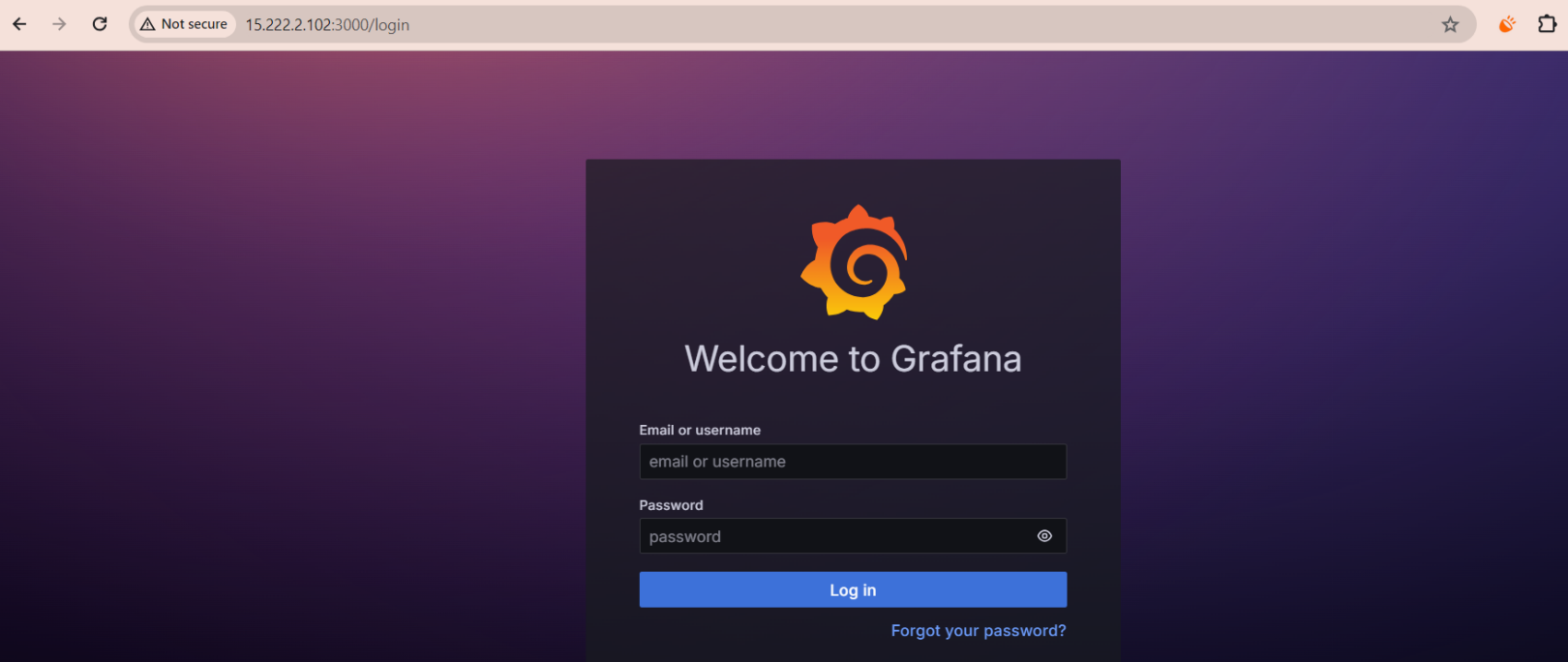
### NOT starting on installation, please execute the following statements to configure grafana to start automatically using systemd

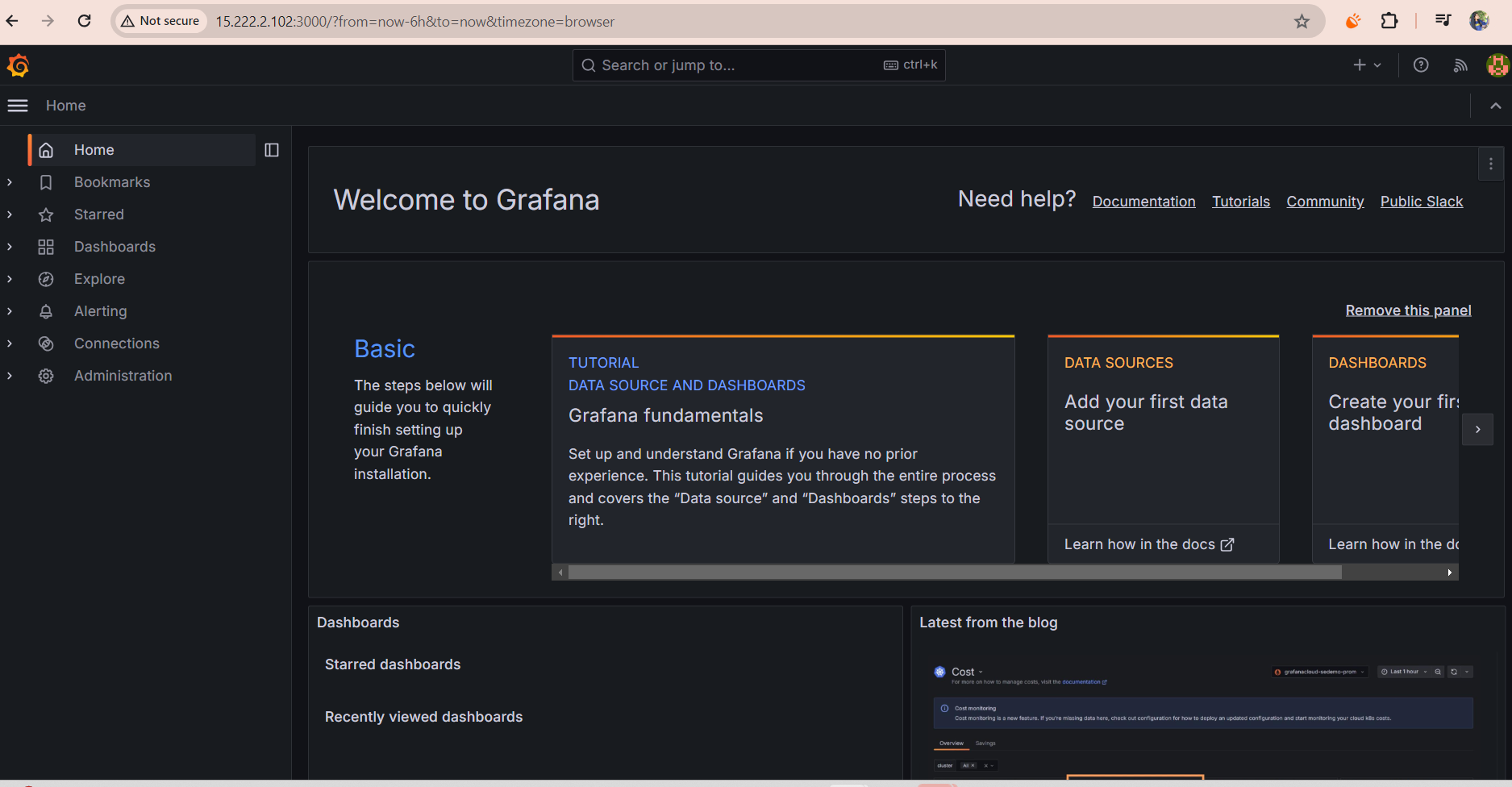
sudo /bin/systemctl daemon-reload

sudo /bin/systemctl enable grafana-server

### You can start grafana-server by executing

sudo /bin/systemctl start grafana-server

* sudo /bin/systemctl start grafana-server
* Default port for Grafana is 3000

Use default creds: admin:admin

***Step4: Install Black box and Node exporter:***

Link: https://prometheus.io/download/#blackbox\_exporter

**Black Box installation:**

* wget <https://github.com/prometheus/blackbox_exporter/releases/download/v0.25.0/blackbox_exporter-0.25.0.linux-amd64.tar.gz>
* tar -xvzf blackbox\_exporter-0.25.0.linux-amd64.tar.gz

blackbox\_exporter-0.25.0.linux-amd64/

blackbox\_exporter-0.25.0.linux-amd64/NOTICE

blackbox\_exporter-0.25.0.linux-amd64/blackbox\_exporter

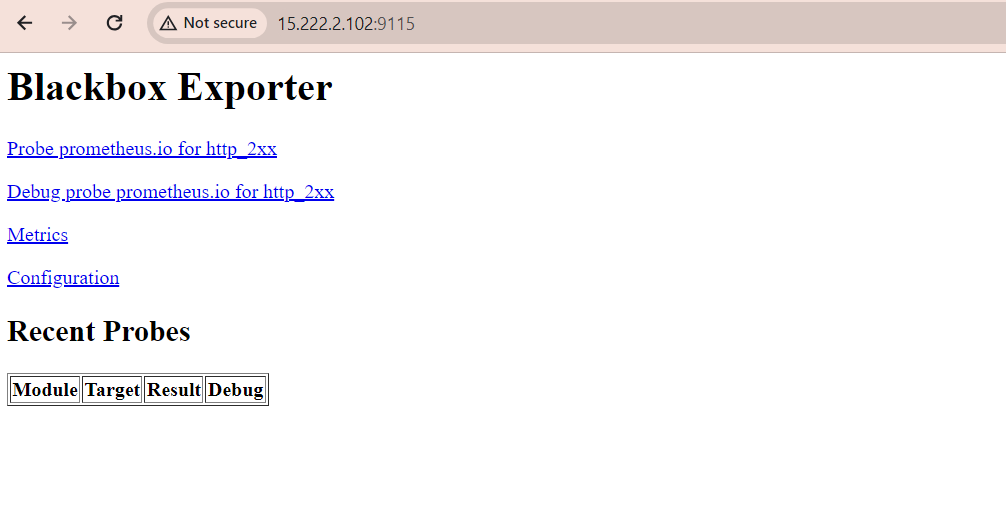
blackbox\_exporter-0.25.0.linux-amd64/LICENSE

blackbox\_exporter-0.25.0.linux-amd64/blackbox.yml

ubuntu@ip-172-31-24-127:~$ ls

* + blackbox\_exporter-0.25.0.linux-amd64 grafana-enterprise\_11.3.0\_amd64.deb prometheus-2.55.0.linux-amd64.tar.gz

blackbox\_exporter-0.25.0.linux-amd64.tar.gz prometheus-2.55.0.linux-amd64

* ls
  + LICENSE NOTICE blackbox.yml blackbox\_exporter
* ./blackbox\_exporter &
*  Use monito server ip on port 9115:
* Update the Prometheus.yaml file with:

scrape\_configs:

- job\_name: 'blackbox'

metrics\_path: /probe

params:

module: [http\_2xx] # Look for a HTTP 200 response.

static\_configs:

- targets:

- http://prometheus.io # Target to probe with http.

- http://example.com:8080 # Target to probe with http on port 8080**. Replace this with the application url**

relabel\_configs:

- source\_labels: [\_\_address\_\_]

target\_label: \_\_param\_target

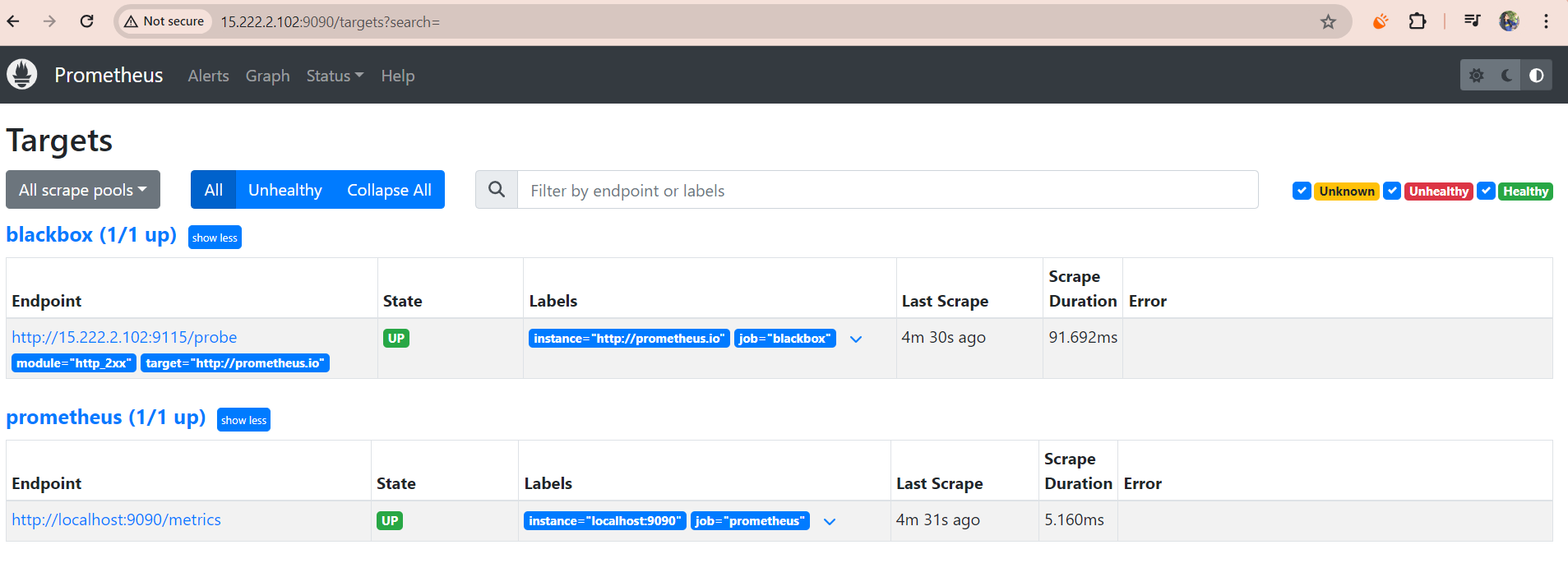
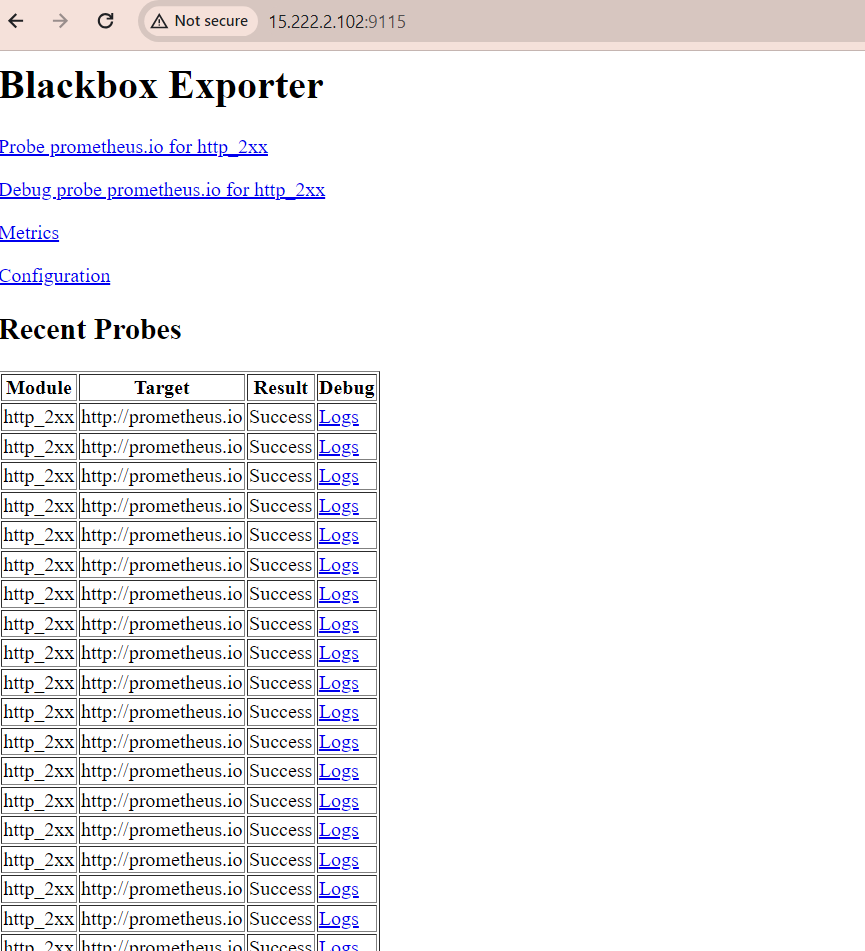
- source\_labels: [\_\_param\_target]

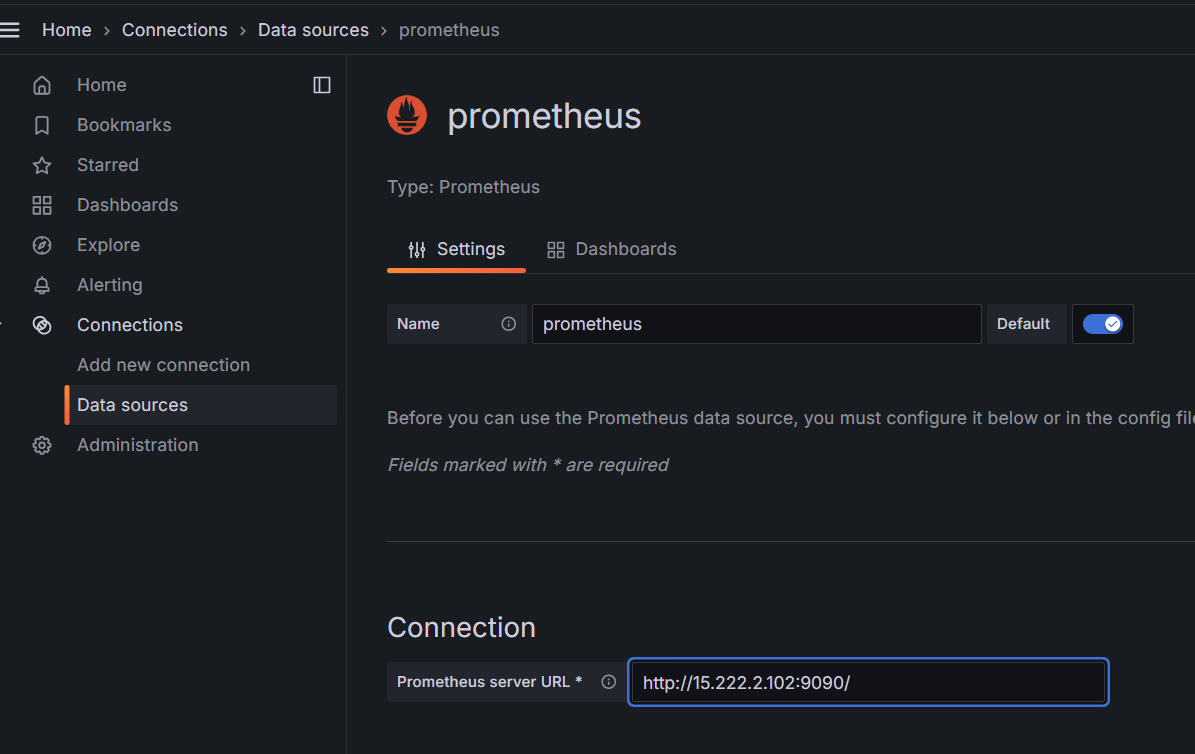
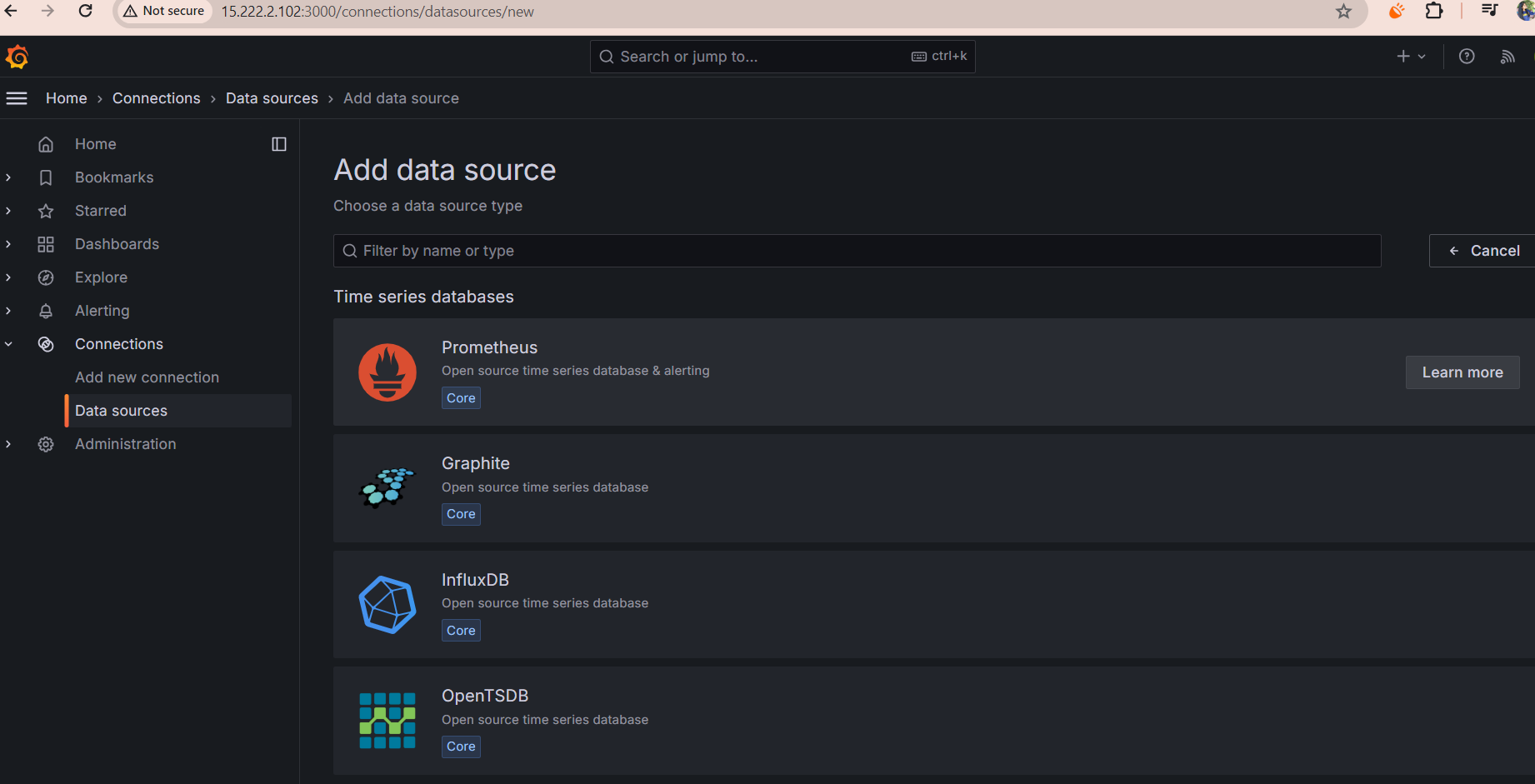
target\_label: instance

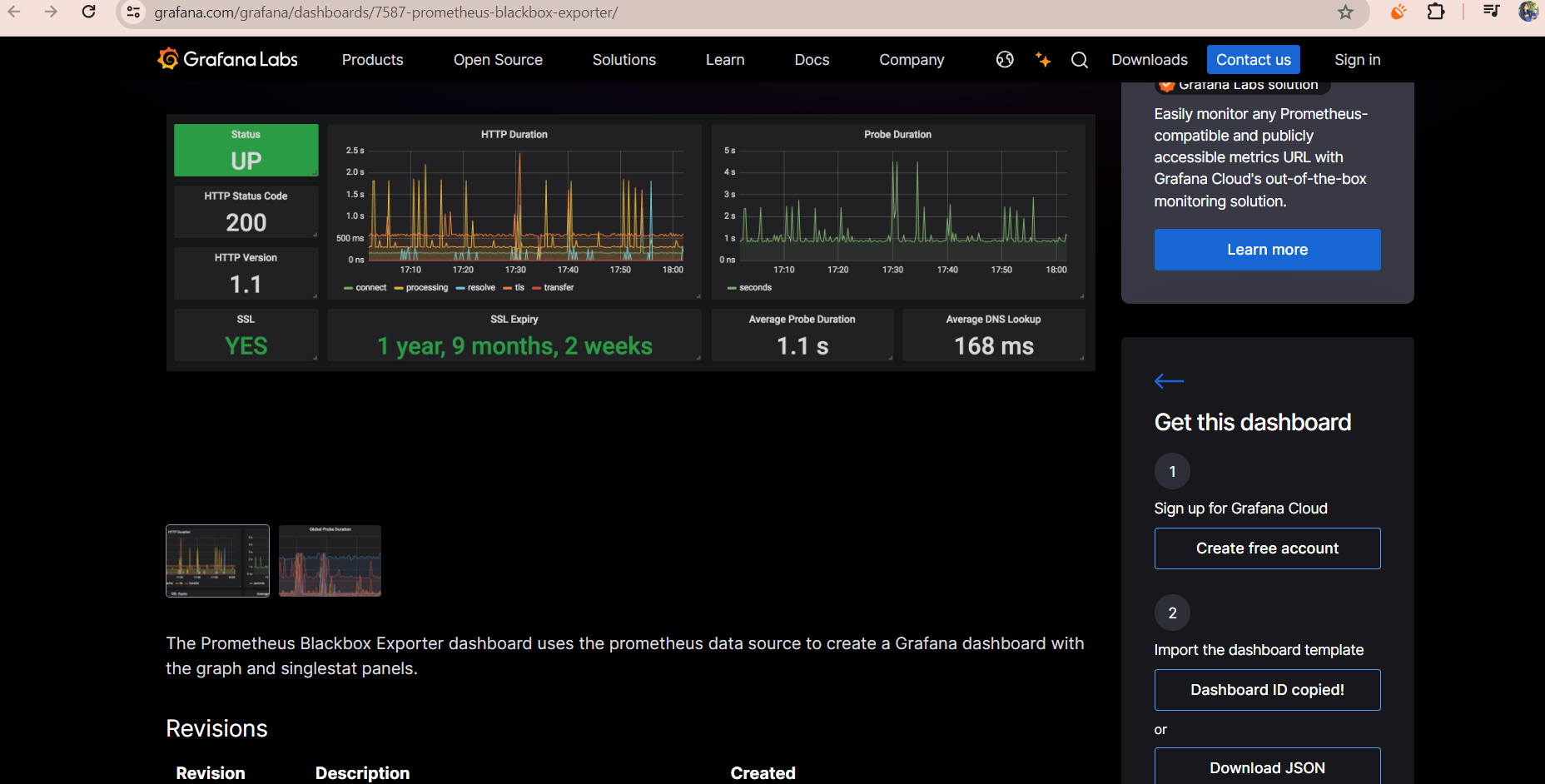
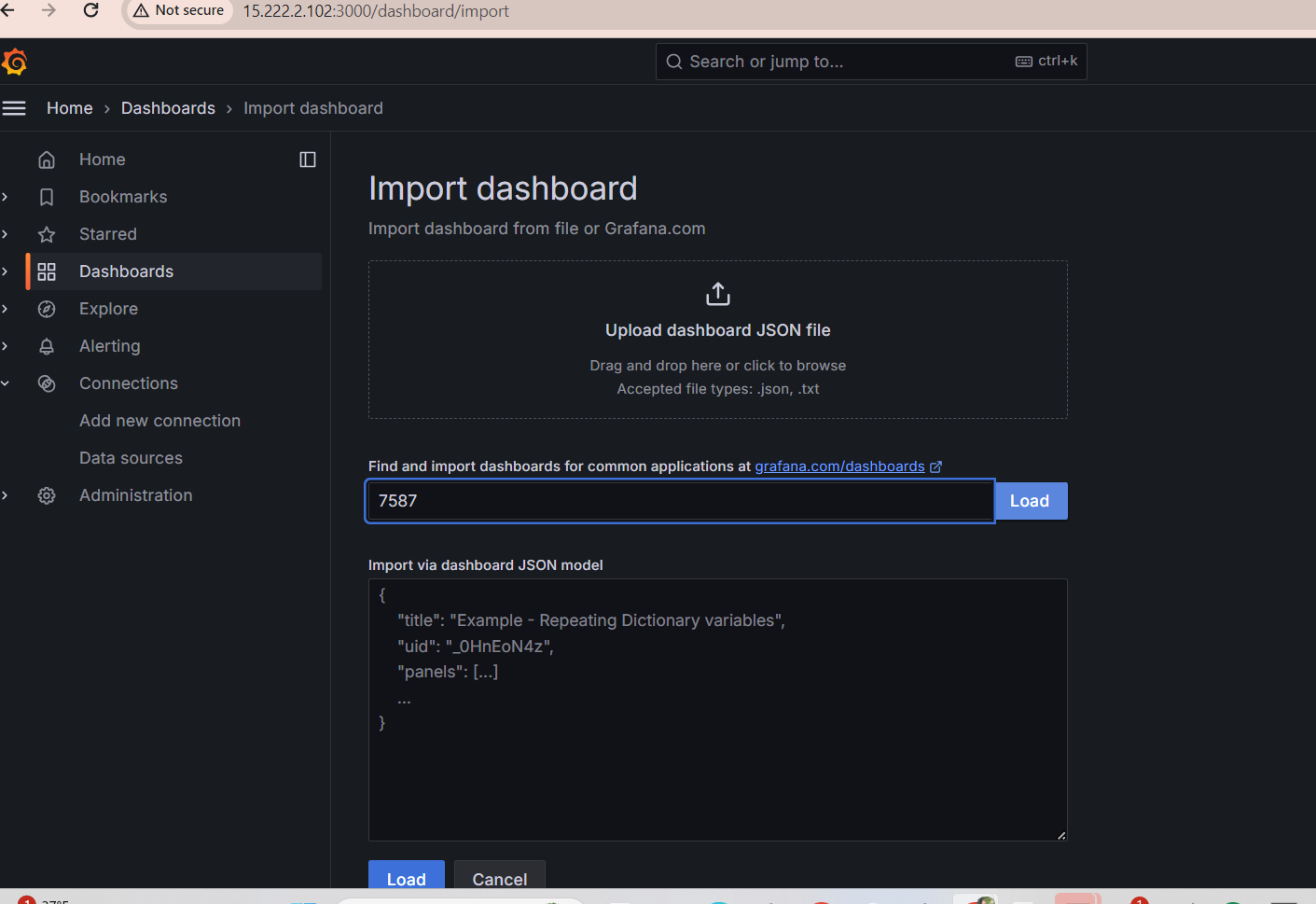
- target\_label: \_\_address\_\_

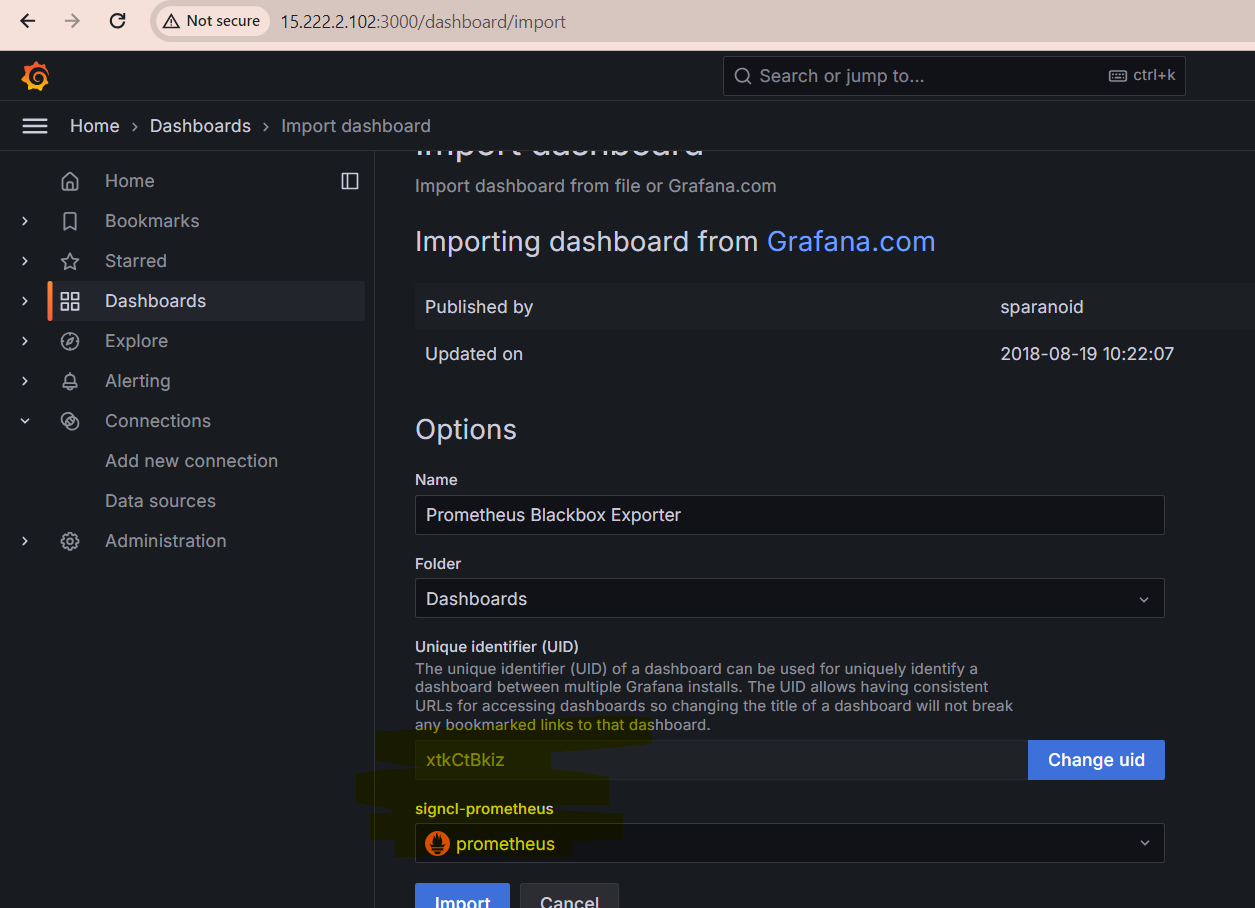
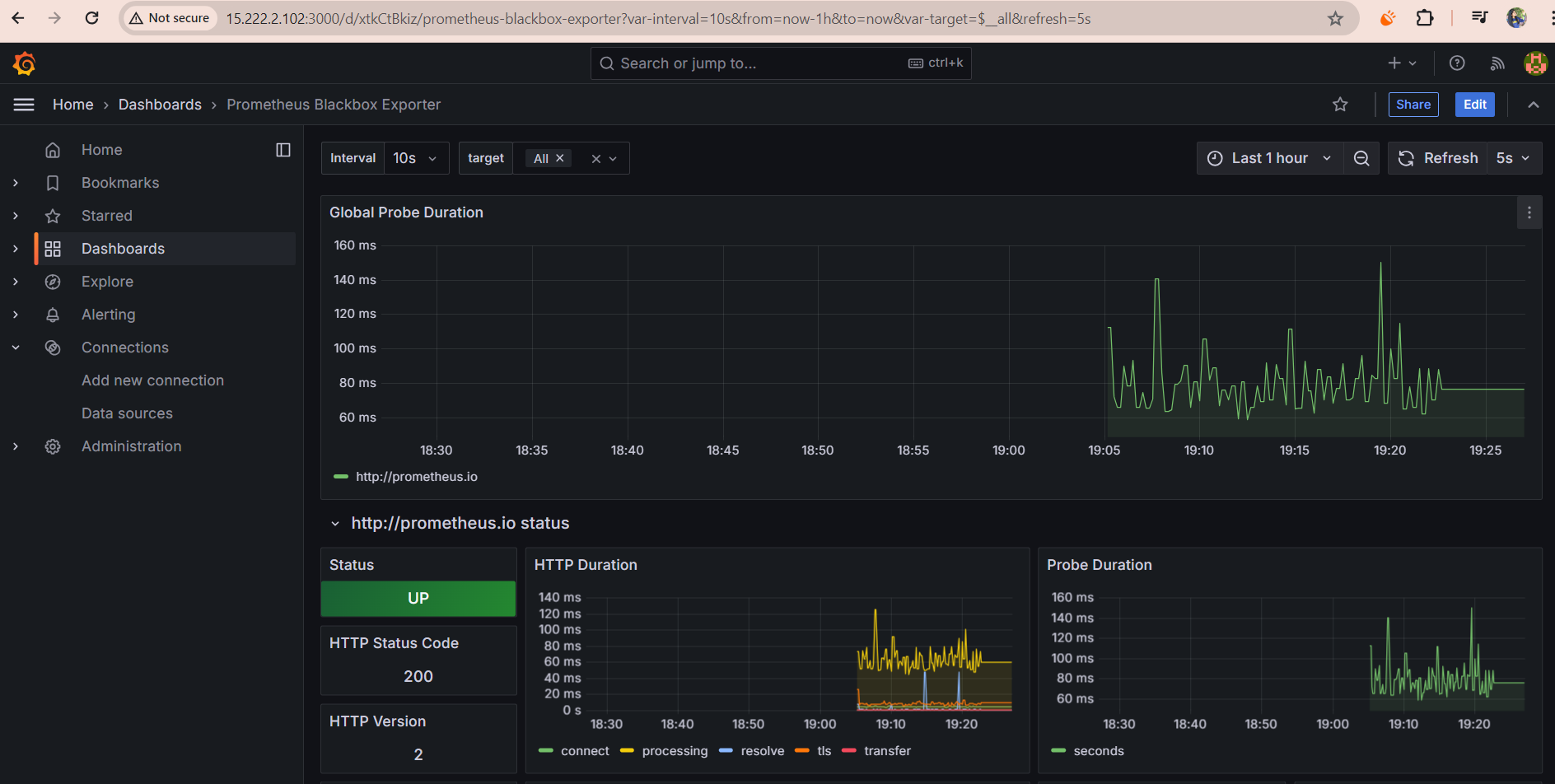
replacement: 15.222.2.102:9115 # The blackbox exporter's real hostname:port.

---The IP is the blackbox server IP

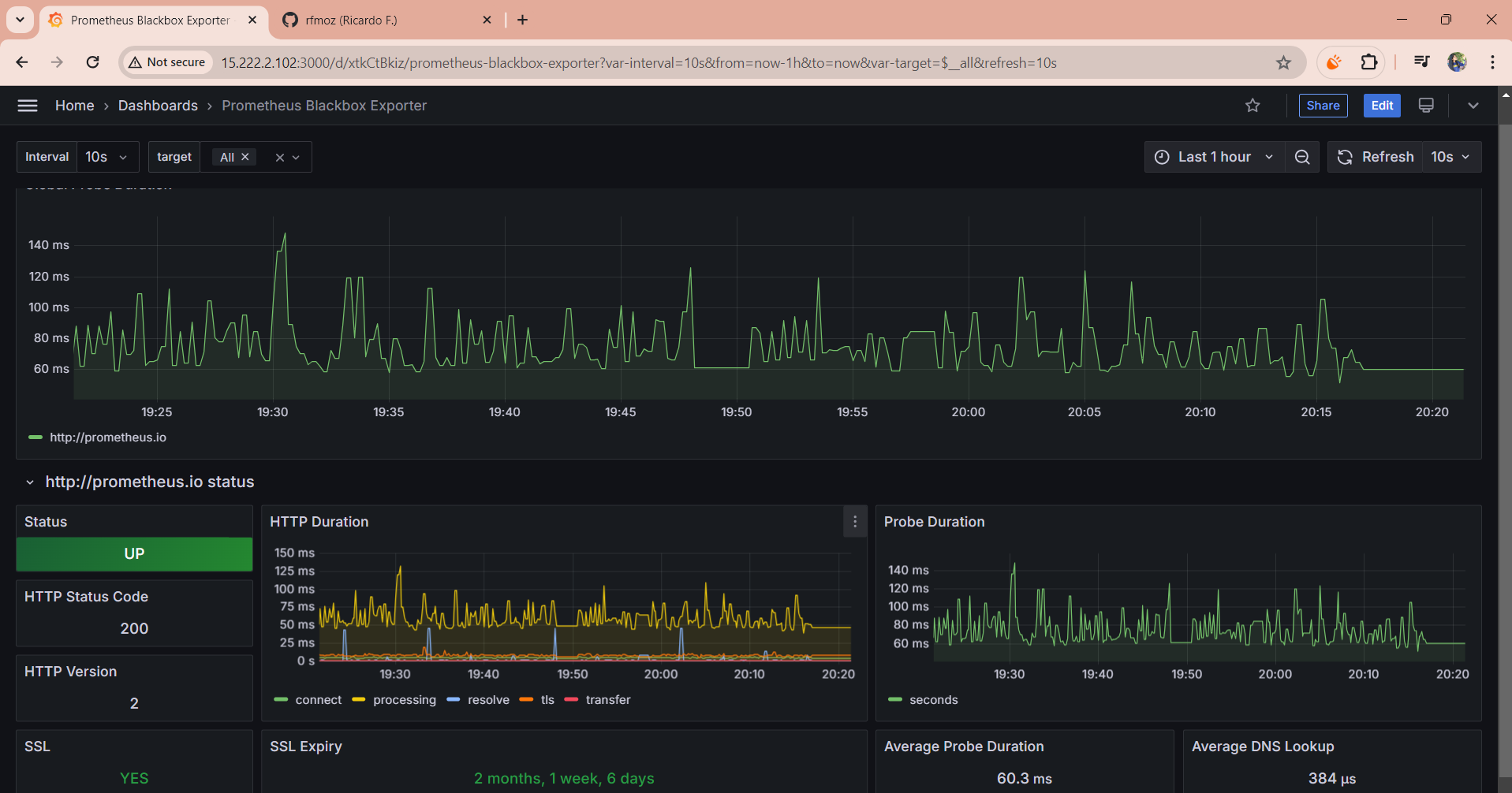
* Now restart Prometheus
* Pgrep Prometheus to get the id
* Kill id
* Restart—
* ./Prometheus &
* After restarting Prometheus go to Prometheus url and referesh it and Go to targets and see if the application that we want to monitor and Prometheus targets that we have set in yaml file are up:
* And under Blackbox Exporter recent probes info is updated:
* Next we need to add Prometheus as data source under Grafana:
* Grafana UI> Datasources>add data source>add Prometheus>Prometheus url-



* Save and test
* Next import dashboard from browser, For this search ‘blackbox grafana dashboard’ in chrome browser and use the Prometheus blackbox exporter > grafana labs link
* Copy the Dashboard ID- 7587
* Now under Grafana UI>Dashboard>Import dashboar

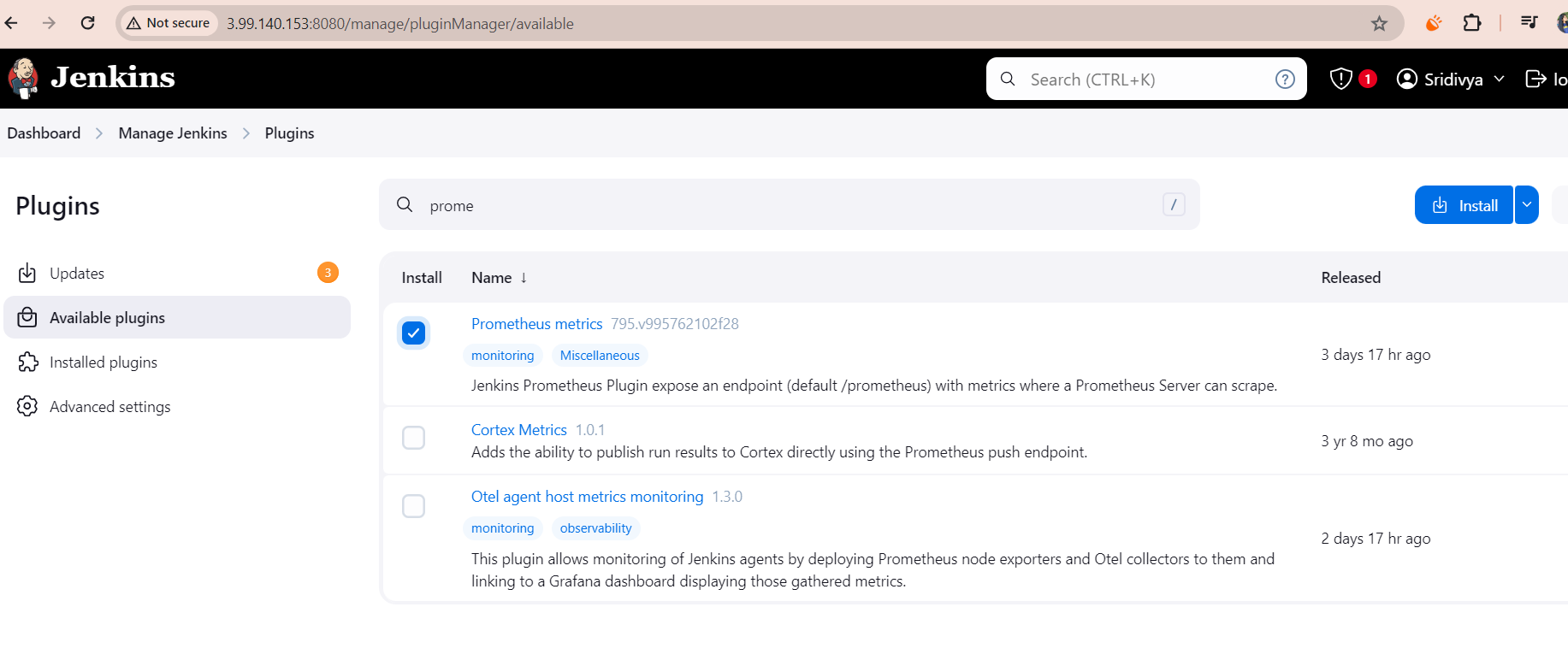


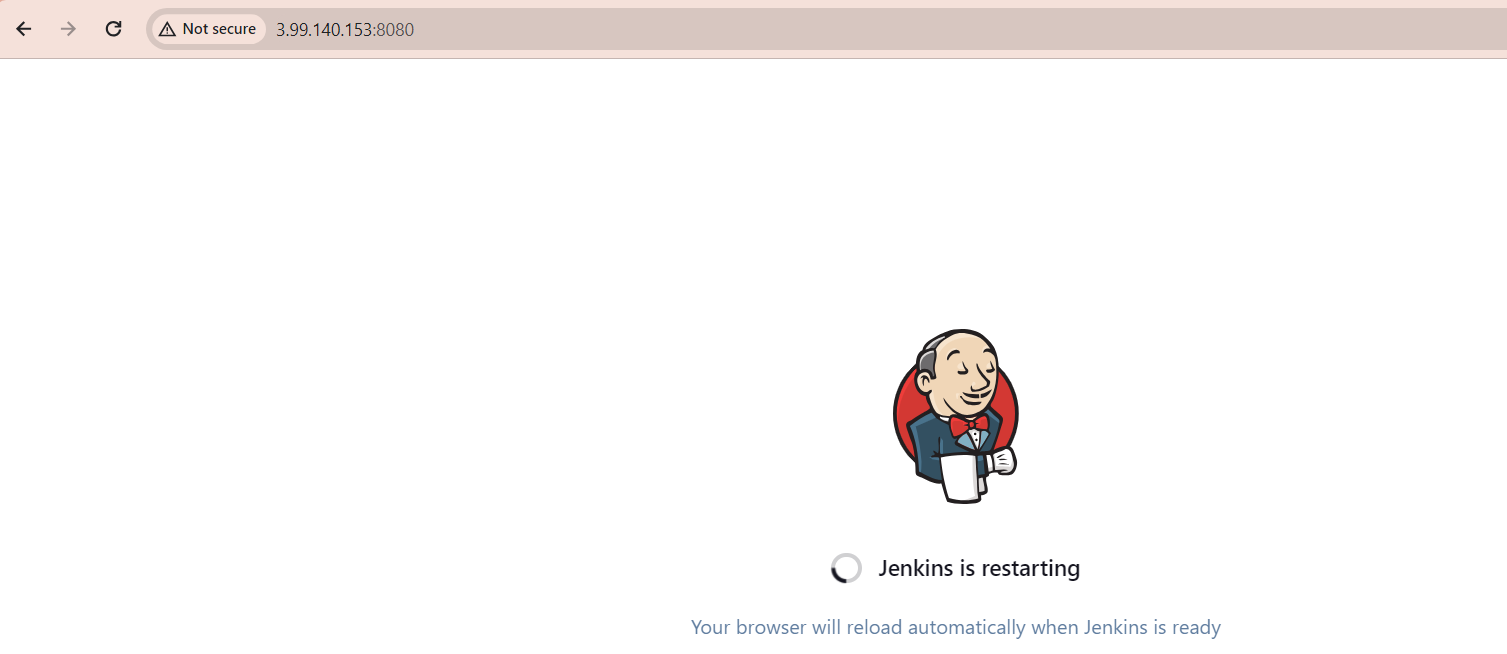
* We can select the targets that we want to monitor in dashboard:



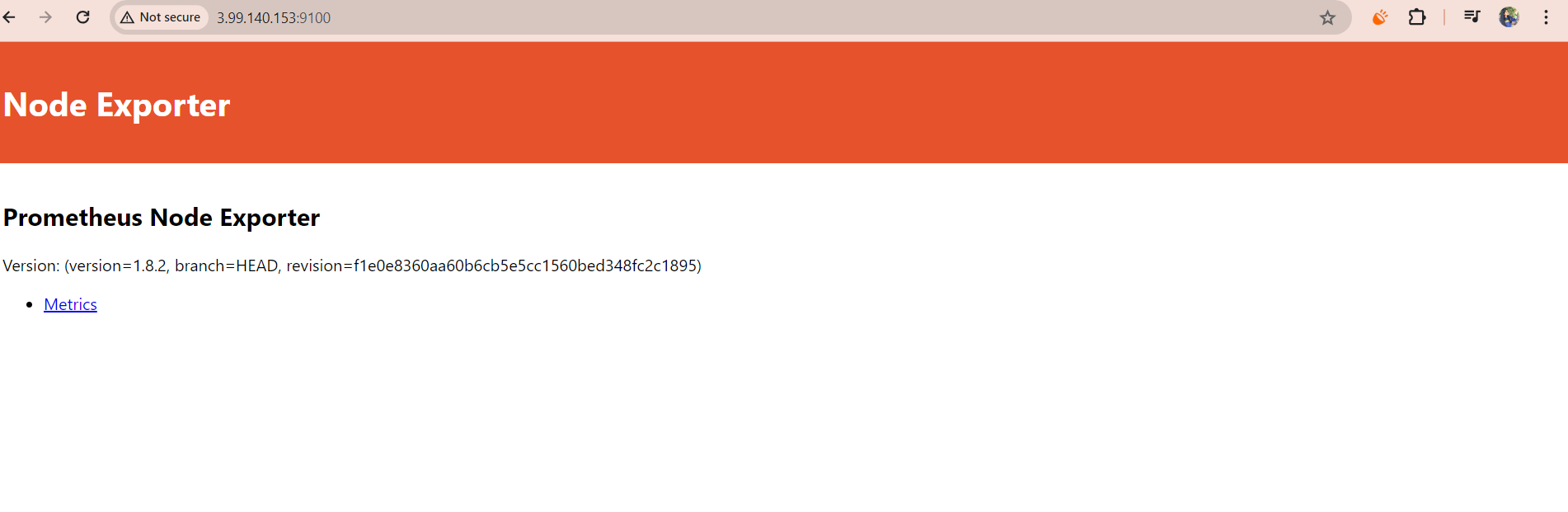
So this is simple Http website monitoring done using blackbox exporter

***Set up System level monitoring, to monitor Jenkins:***

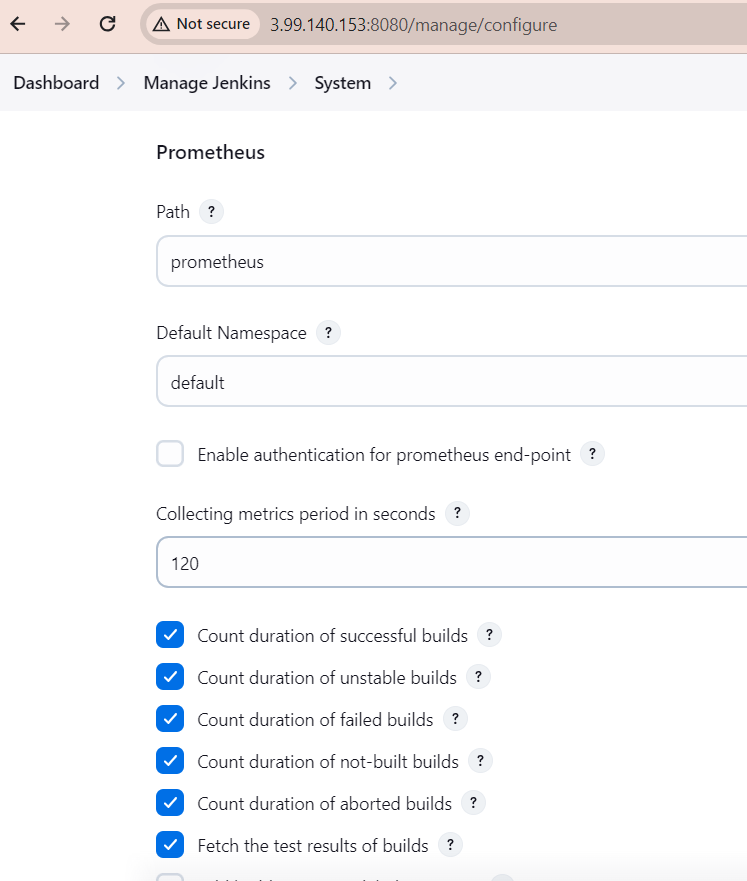
* For this lets first install Prometheus plugin and restart the Jenkins: IP:8080/restart:

***Install Node exporter from below link on Jenkins server:***

* wget <https://github.com/prometheus/node_exporter/releases/download/v1.8.2/node_exporter-1.8.2.linux-amd64.tar.gz>
* root@ip-172-31-18-47:~# tar -xvf node\_exporter-1.8.2.linux-amd64.tar.gz
* run it -- ./node\_exporter &
* to access node exporter use the Jenkins ip and default port:9100



***Now in Jenkins browser we can see the Prometheus settings:***



* Configure Prometheus settings in Monitor Server by updating Prometheus.yml file, with Jenkins and node expoerter job info as :

scrape\_configs:

# The job name is added as a label `job=<job\_name>` to any timeseries scraped from this config.

- job\_name: "prometheus"

# metrics\_path defaults to '/metrics'

# scheme defaults to 'http'.

static\_configs:

- targets: ["localhost:9090"]

- job\_name: "node\_exporter"

# metrics\_path defaults to '/metrics'

# scheme defaults to 'http'.

static\_configs:

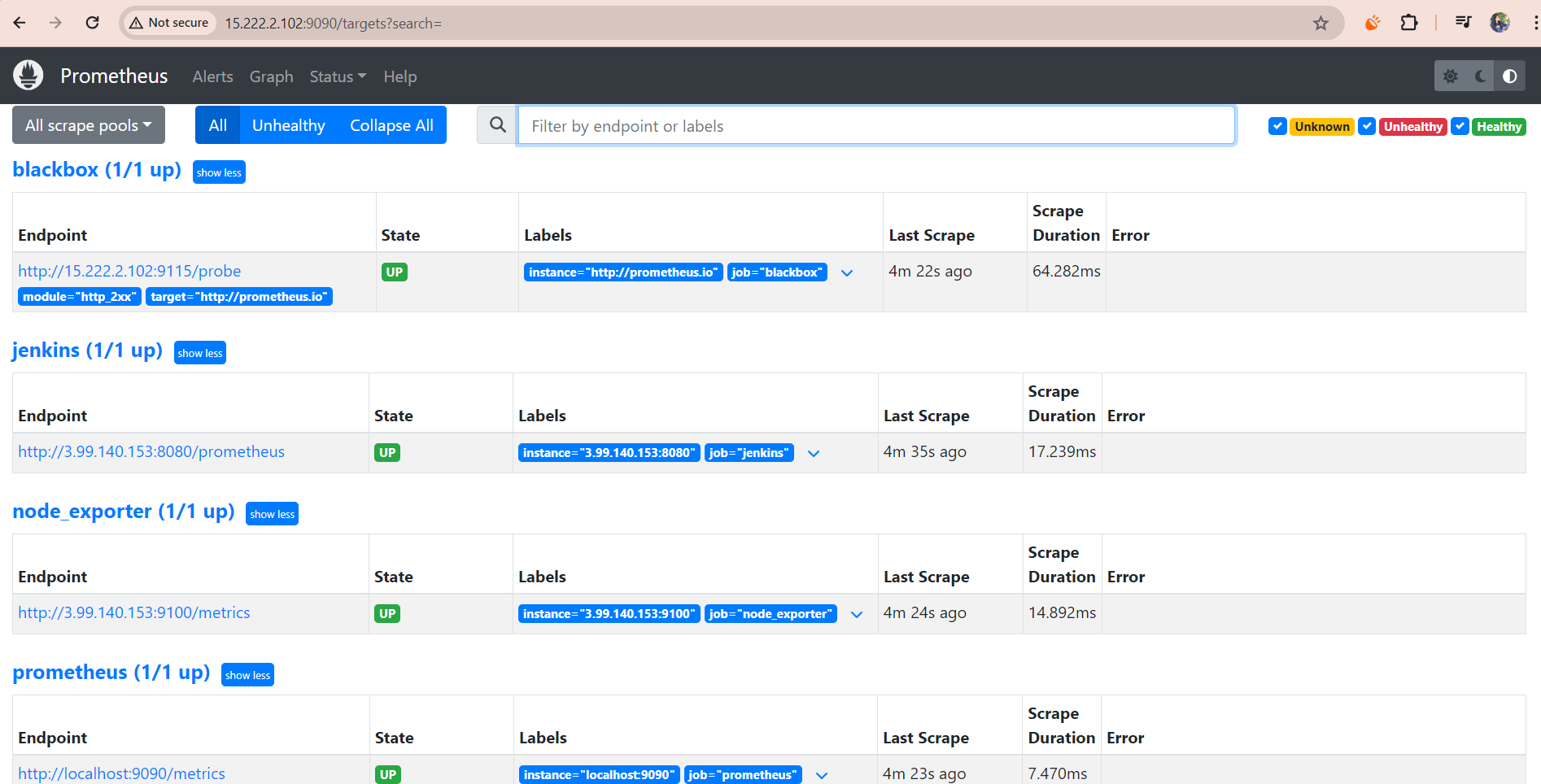
- targets: ["3.99.140.153:9100"]

- job\_name: 'jenkins'

metrics\_path: '/prometheus'

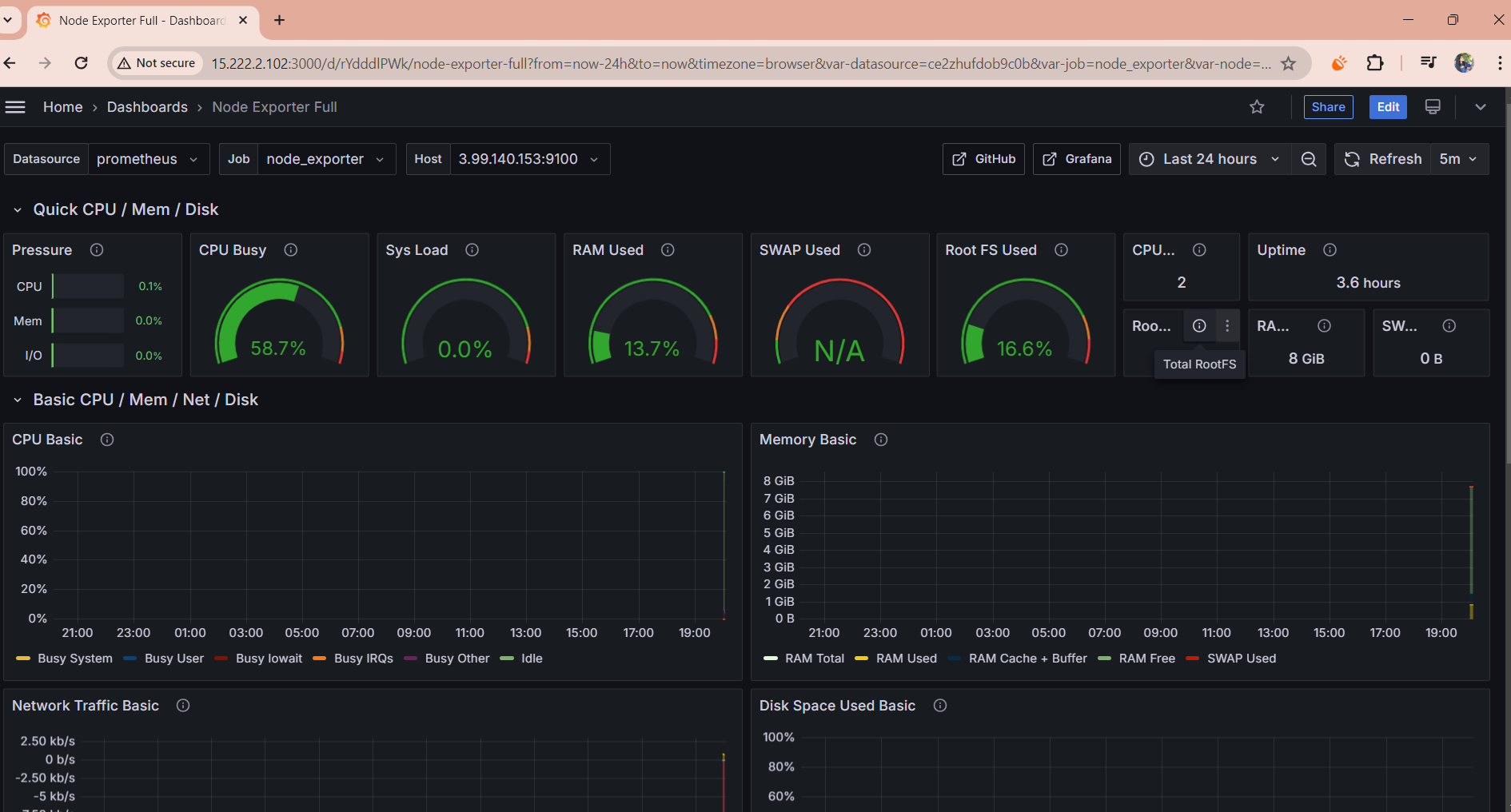
static\_configs:

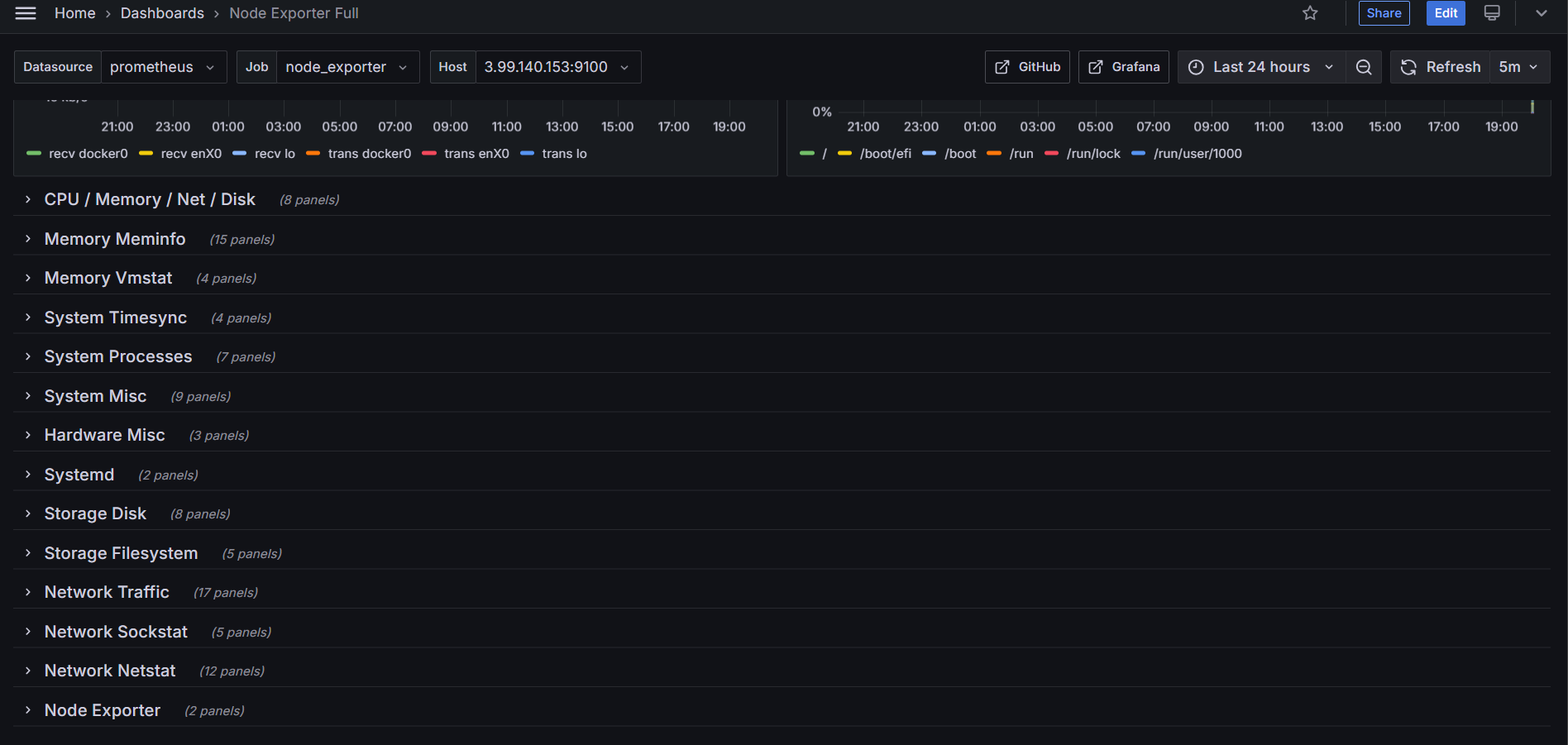
- targets: ["3.99.140.153:8080"]

***Now we Node exporter and Jenkins can be seen under Prometheus targets:***

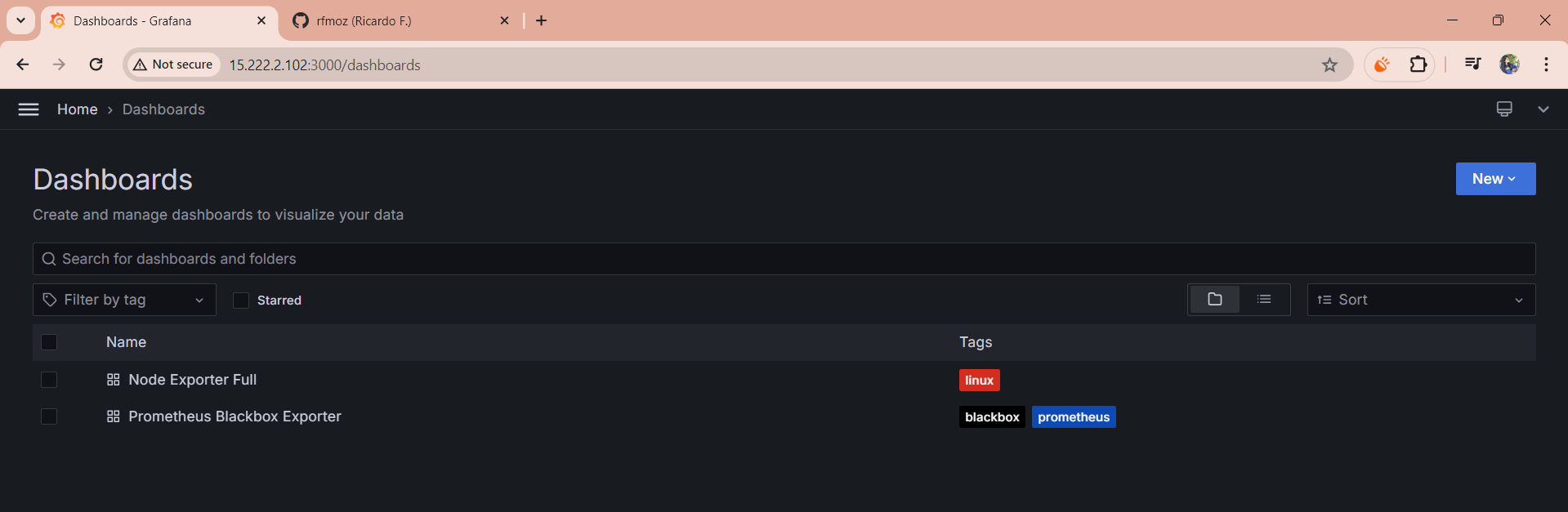
***Step5: Create Dashboard in Grafana to monitor Jenkins***

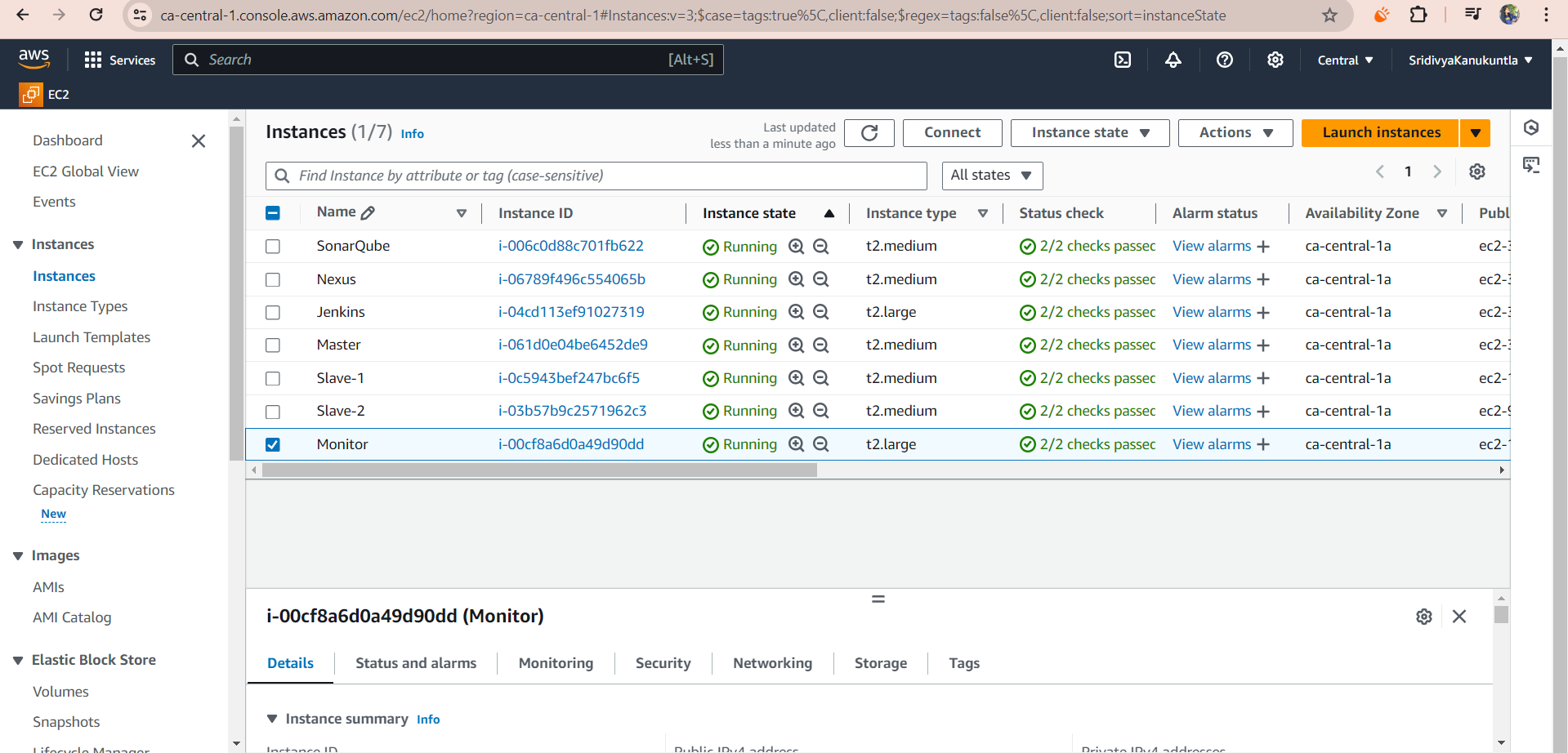
* Search for node exporter Grafana dashboard in chrome
* And copy the ID and import it in Grafana, as we did earlier: 1860:





***Here are the 2 dashboards created:***



***The overview of the instances created for the project:***