Set up Grafana , Prometheus and Blackbox Exporter

Table of Content

1. Definition of Grafana , Prometheus and Blackbox Exporter

a. Grafana: Grafana is an open-source analytics and monitoring platform that integrates with various data sources, allowing users to visualize and understand metrics through customizable dashboards.

b. Prometheus: Prometheus is an open-source monitoring and alerting toolkit designed for reliability and scalability. It collects and stores time-series data, offering powerful querying and alerting capabilities.

c. Blackbox Exporter: Blackbox Exporter is a Prometheus exporter designed for probing and monitoring external services. It allows users to check the availability and response of endpoints, such as HTTP, TCP, ICMP, and DNS, and generates metrics based on the results.

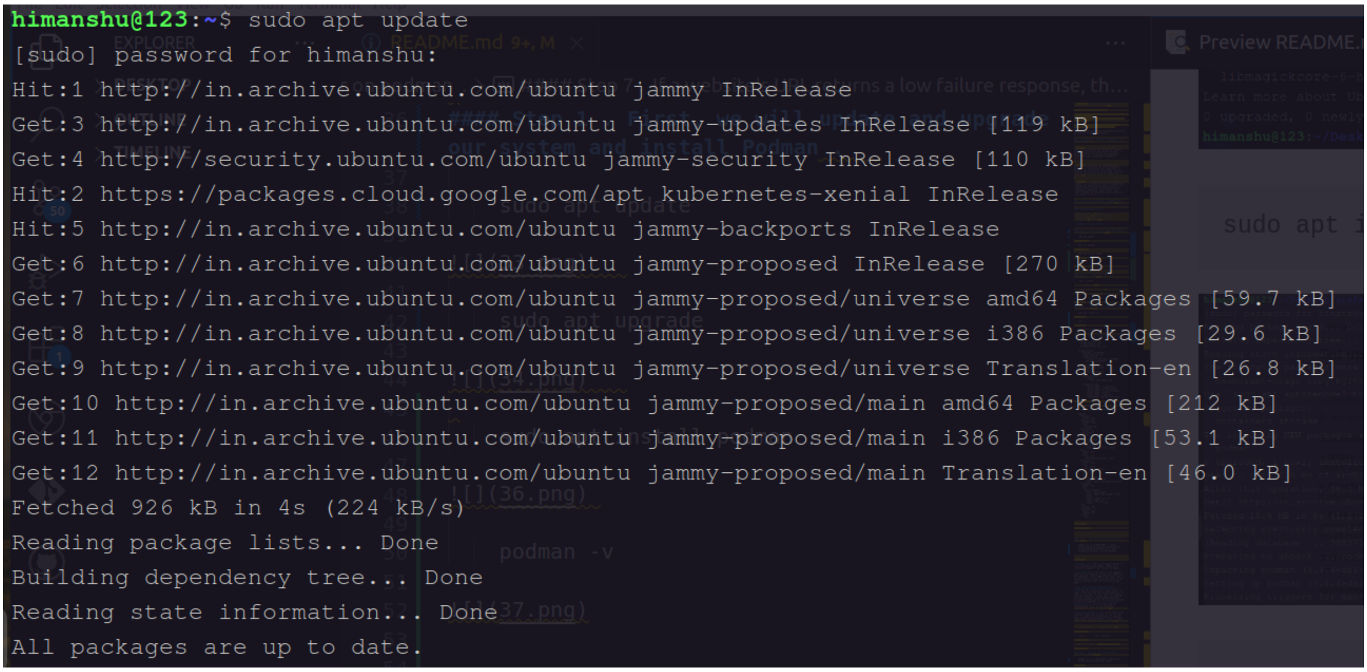
2. How to set up Grafana , Prometheus and Blackbox Exporter on Podman Container ?

System Requirement

- **Distributor ID: Ubuntu Description: Ubuntu 22.04.3 LTS Release: 22.04 Codename: jammy**
- **podman version 3.4.4**

Step 1 . First, we will update and upgrade our system and install Podman .

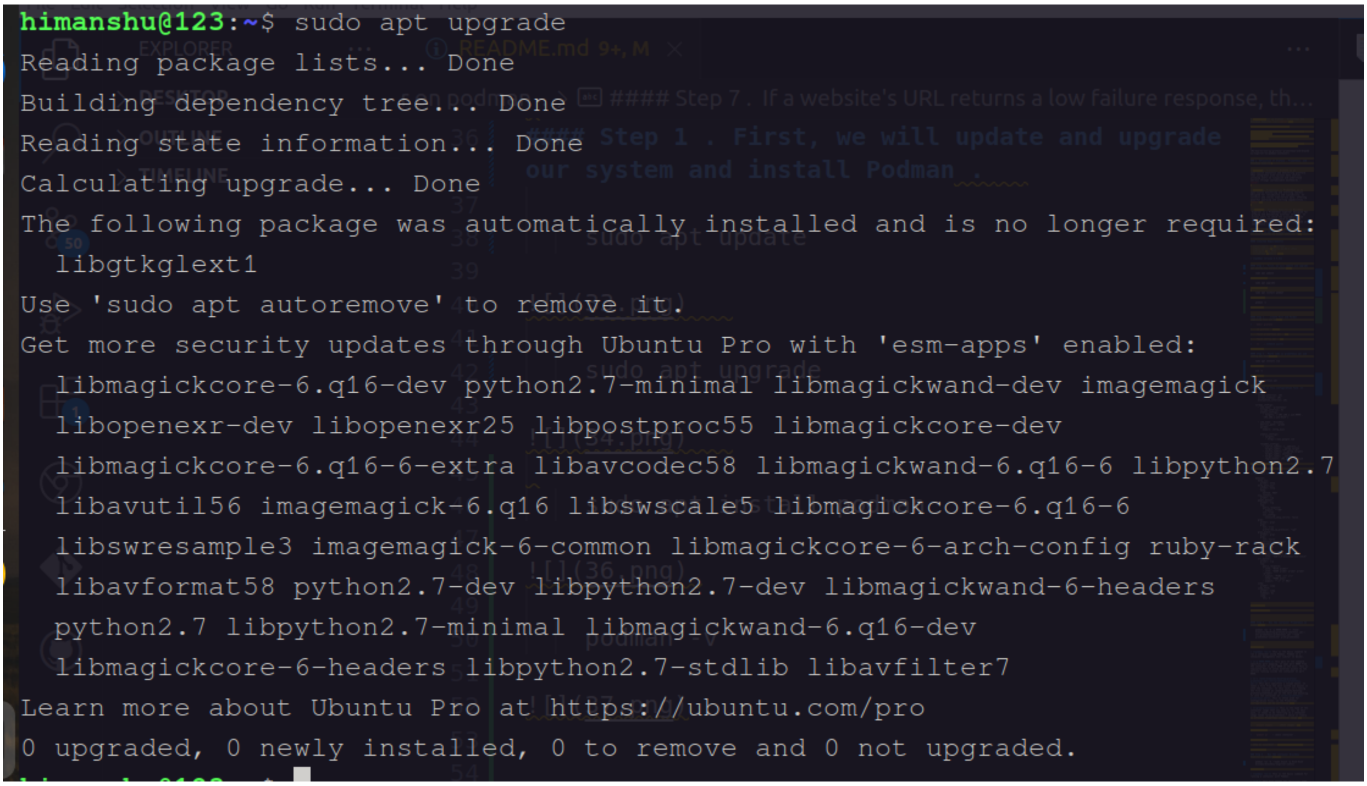
```
sudo apt update
```



```
himanshu@123:~$ sudo apt update
[sudo] password for himanshu:
Hit:1 http://in.archive.ubuntu.com/ubuntu jammy InRelease
Get:3 http://in.archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Get:4 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Hit:2 https://packages.cloud.google.com/apt kubernetes-xenial InRelease
Hit:5 http://in.archive.ubuntu.com/ubuntu jammy-backports InRelease
Get:6 http://in.archive.ubuntu.com/ubuntu jammy-proposed InRelease [270 kB]
Get:7 http://in.archive.ubuntu.com/ubuntu jammy-proposed/universe amd64 Packages [59.7 kB]
Get:8 http://in.archive.ubuntu.com/ubuntu jammy-proposed/universe i386 Packages [29.6 kB]
Get:9 http://in.archive.ubuntu.com/ubuntu jammy-proposed/universe Translation-en [26.8 kB]
Get:10 http://in.archive.ubuntu.com/ubuntu jammy-proposed/main amd64 Packages [212 kB]
Get:11 http://in.archive.ubuntu.com/ubuntu jammy-proposed/main i386 Packages [53.1 kB]
Get:12 http://in.archive.ubuntu.com/ubuntu jammy-proposed/main Translation-en [46.0 kB]
Fetched 926 kB in 4s (224 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
All packages are up to date.
```

- **sudo:** This part of the command gives you temporary administrative privileges. It stands for "superuser do." It allows you to execute commands that require special permissions, like updating software.
- **apt:** This stands for "Advanced Package Tool." It's a package management system used in many Linux distributions, including Ubuntu and Debian. It helps you install, update, and remove software on your system.
- **update:** This is the action you're telling APT to perform. When you run sudo apt update, you're asking the system to update its list of available software packages. It checks the internet for any updates to the packages installed on your system and retrieves this information.
- **System Upgrade**

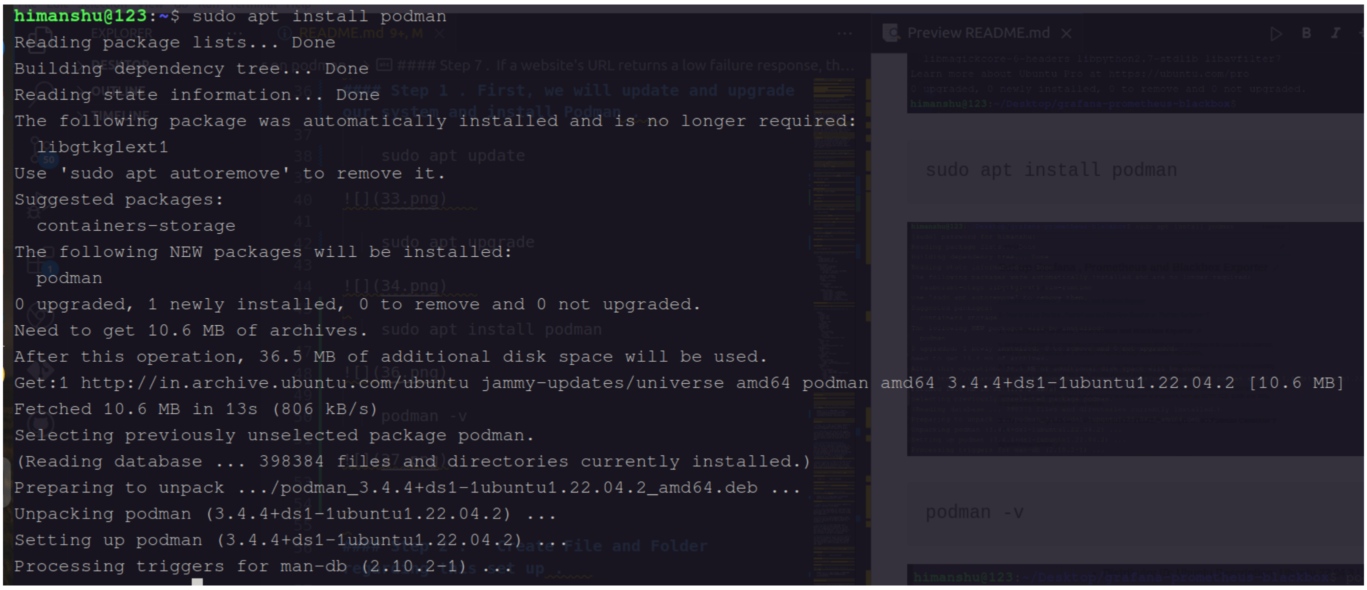
  sudo apt upgrade



```
himanshu@123:~$ sudo apt upgrade
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Calculating upgrade... Done
The following package was automatically installed and is no longer required:
  libgtkglext1
Use 'sudo apt autoremove' to remove it.
Get more security updates through Ubuntu Pro with 'esm-apps' enabled:
  libmagickcore-6.q16-dev python2.7-minimal libmagickwand-dev imagemagick
  libopenexr-dev libopenexr25 libpostproc55 libmagickcore-dev
  libmagickcore-6.q16-6-extra libavcodec58 libmagickwand-6.q16-6 libpython2.7
  libavutil56 imagemagick-6.q16 libswscale5 libmagickcore-6.q16-6
  libswresample3 imagemagick-6-common libmagickcore-6-arch-config ruby-rack
  libavformat58 python2.7-dev libpython2.7-dev libmagickwand-6-headers
  python2.7 libpython2.7-minimal libmagickwand-6.q16-dev
  libmagickcore-6-headers libpython2.7-stdlib libavfilter7
Learn more about Ubuntu Pro at https://ubuntu.com/pro
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
```

- **sudo:** This stands for "superuser do" and it's a way to tell the system that you want to perform the following command with administrator privileges. In other words, you're asking for permission to make changes to your system, and it's usually used for tasks that require special permissions.
- **apt:** This is a package manager for Debian-based Linux distributions like Ubuntu. It's a tool that helps you install, update, and manage software packages on your computer. You can think of it as a way to easily install and update programs.
- **upgrade:** This is the action you're telling apt to perform. When you run sudo apt upgrade, you're instructing the package manager to check for updates for all the software packages currently installed on your system.
- **Install Podman**

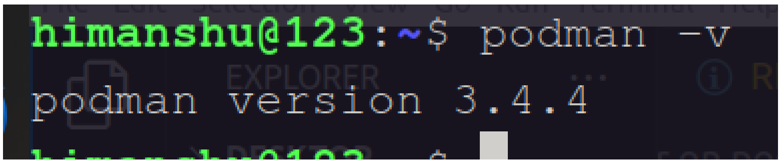
  sudo apt install podman



```
himanshu@123:~$ sudo apt install podman
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following package was automatically installed and is no longer required:
  libgtkglext1
Use 'sudo apt autoremove' to remove it.
Suggested packages:
  containers-storage
The following NEW packages will be installed:
  podman
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 10.6 MB of archives.
After this operation, 36.5 MB of additional disk space will be used.
Get:1 http://in.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 podman amd64 3.4.4+ds1-1ubuntu1.22.04.2 [10.6 MB]
Fetched 10.6 MB in 13s (806 kB/s)
Selecting previously unselected package podman.
(Reading database ... 398384 files and directories currently installed.)
Preparing to unpack .../podman_3.4.4+ds1-1ubuntu1.22.04.2_amd64.deb ...
Unpacking podman (3.4.4+ds1-1ubuntu1.22.04.2) ...
Setting up podman (3.4.4+ds1-1ubuntu1.22.04.2) ...
Processing triggers for man-db (2.10.2-1) ...
```

- **sudo:** This is a command that stands for "superuser do." It's used to execute the following command with administrative privileges. In other words, it allows you to make system-wide changes and install software.
- **apt:** This is a package manager for Debian-based Linux distributions like Ubuntu. It's a tool for managing software packages, including installation, updates, and removal.
- **install:** This is the action you want to perform with apt. You're telling it to install a software package.
- **podman:** This is the name of the software package you want to install. Podman is a container management tool, similar to Docker, which allows you to run and manage containers on your system.
- **Check Version**

  podman -v

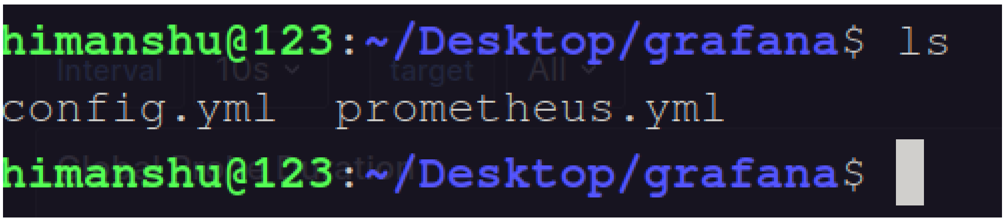


```
himanshu@123:~$ podman -v
podman version 3.4.4
```

- **Podman:** This is the name of the software or program you are using.
- **-v:** This is an option or flag that you add to the command to tell Podman to show you the version information.

Step 2 . Create File and Folder regarding this set up .

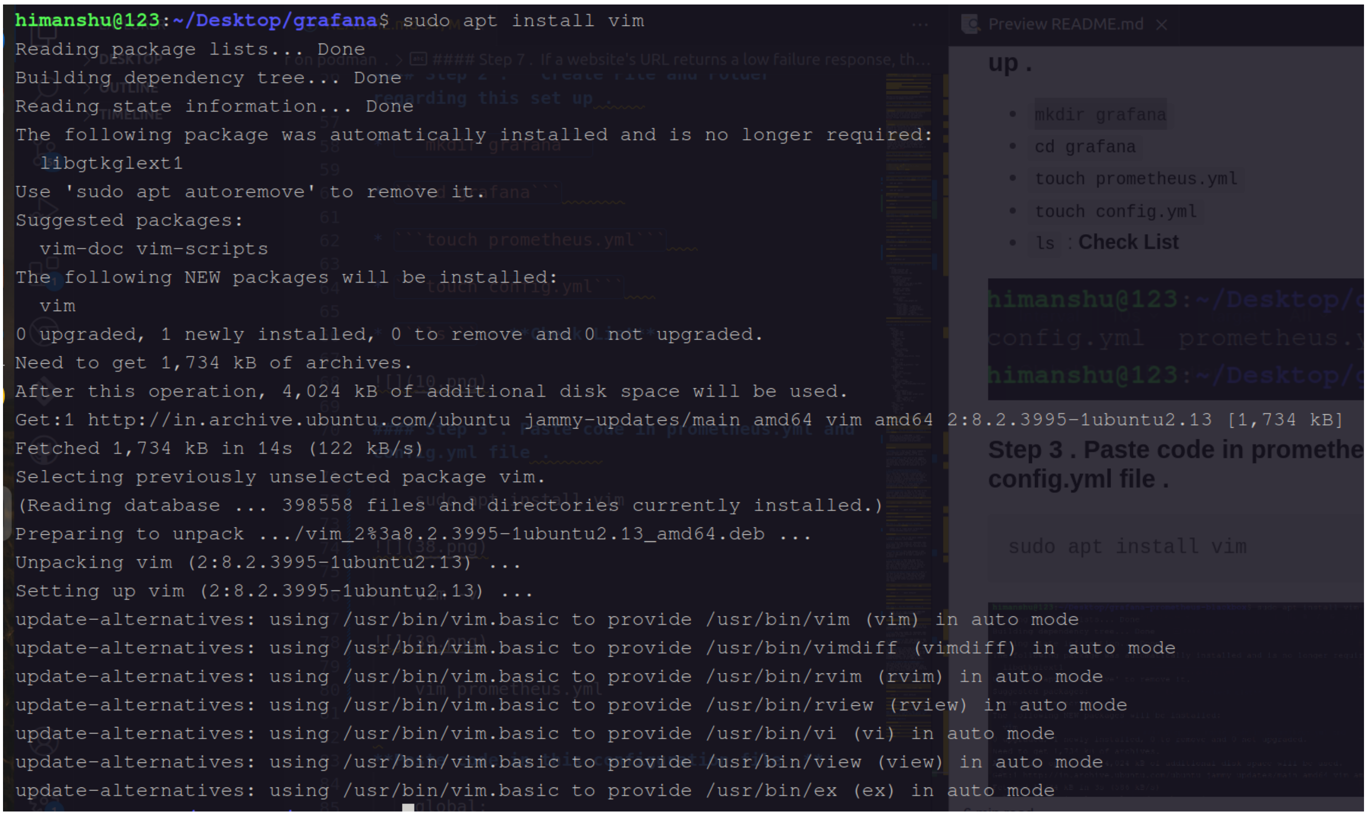
- `cd Desktop`
- `mkdir grafana`
- `cd grafana`
- `touch prometheus.yml`
- `touch config.yml`
- `ls` : **Check List**



```
himanshu@123:~/Desktop/grafana$ ls
config.yml  prometheus.yml
himanshu@123:~/Desktop/grafana$
```

Step 3 . Paste code in prometheus.yml and config.yml file .

```
sudo apt install vim
```



```
himanshu@123:~/Desktop/grafana$ sudo apt install vim
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following package was automatically installed and is no longer required:
  libgtkglext1
Use 'sudo apt autoremove' to remove it.
Suggested packages:
  vim-doc vim-scripts
The following NEW packages will be installed:
  vim
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 1,734 kB of archives.
After this operation, 4,024 kB of additional disk space will be used.
Get:1 http://in.archive.ubuntu.com/ubuntu jammy-updates/main amd64 vim amd64 2:8.2.3995-1ubuntu2.13 [1,734 kB]
Fetched 1,734 kB in 14s (122 kB/s)
Selecting previously unselected package vim.
(Reading database ... 398558 files and directories currently installed.)
Preparing to unpack .../vim_2%3a8.2.3995-1ubuntu2.13_amd64.deb ...
Unpacking vim (2:8.2.3995-1ubuntu2.13) ...
Setting up vim (2:8.2.3995-1ubuntu2.13) ...
update-alternatives: using /usr/bin/vim.basic to provide /usr/bin/vim (vim) in auto mode
update-alternatives: using /usr/bin/vim.basic to provide /usr/bin/vimdiff (vimdiff) in auto mode
update-alternatives: using /usr/bin/vim.basic to provide /usr/bin/rvim (rvim) in auto mode
update-alternatives: using /usr/bin/vim.basic to provide /usr/bin/rview (rview) in auto mode
update-alternatives: using /usr/bin/vim.basic to provide /usr/bin/vi (vi) in auto mode
update-alternatives: using /usr/bin/vim.basic to provide /usr/bin/view (view) in auto mode
update-alternatives: using /usr/bin/vim.basic to provide /usr/bin/ex (ex) in auto mode
```

- **sudo:** This part of the command is used to run the following command with superuser privileges. It allows you to make system-wide changes, which are usually restricted to the system administrator.

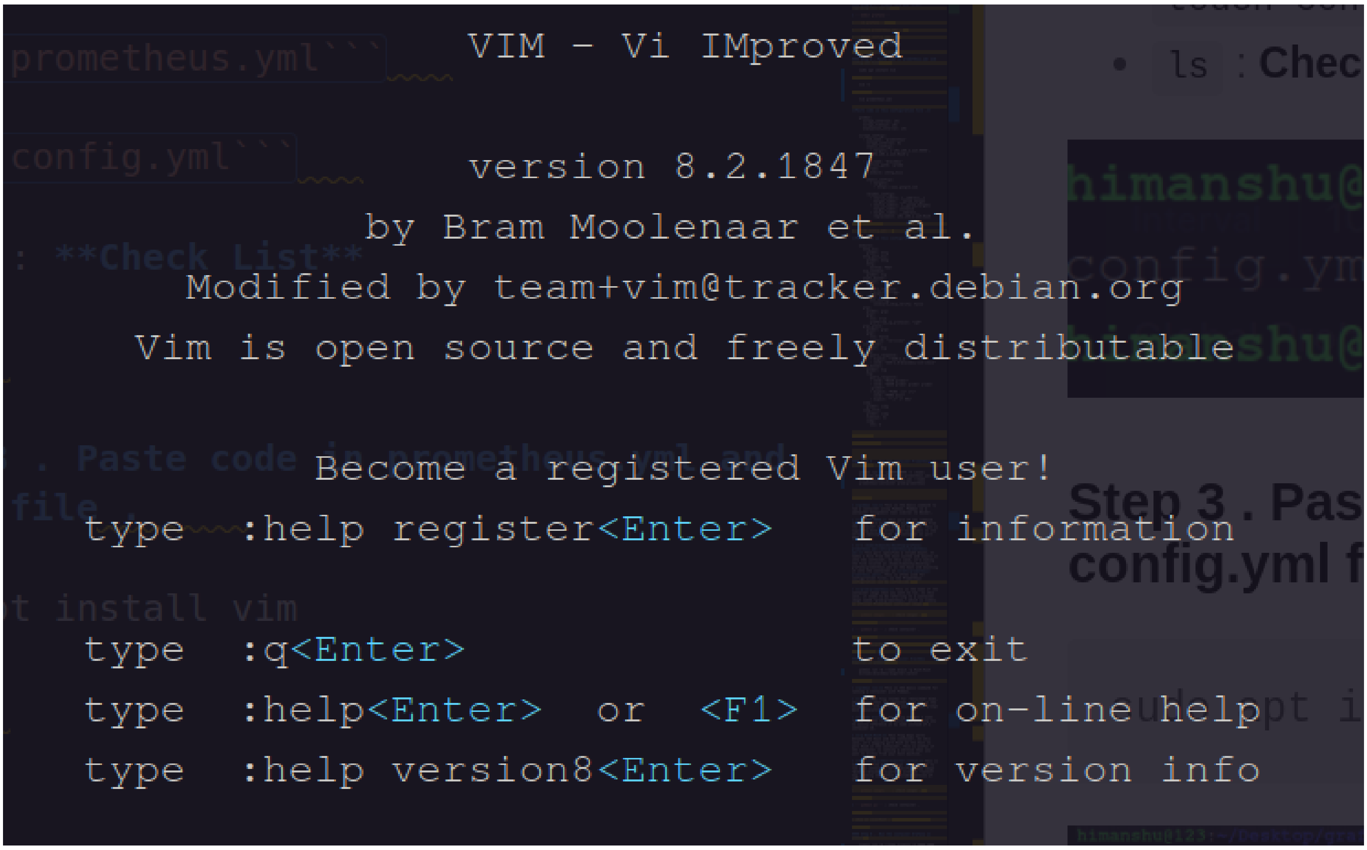
- **apt:** This is a package manager used in Debian-based Linux distributions like Ubuntu. It helps you install, update, and manage software on your computer.

- **install:** This is the action you want to perform with apt. You're telling apt to install a software package.

- **vim:** This is the name of the software package you want to install. Vim is a popular text editor that's often used in the command line.

- **Check Version**

  vim -v



```
                VIM - Vi IMproved

                version 8.2.1847
            by Bram Moolenaar et al.
      Modified by team+vim@tracker.debian.org
    Vim is open source and freely distributable

           Become a registered Vim user!
  type  :help register<Enter>   for information

  type  :q<Enter>               to exit
  type  :help<Enter>  or  <F1>  for on-line help
  type  :help version8<Enter>   for version info
```

- **Vim:** Vim is a powerful and popular text editor that you can use to create and edit text files, code, and various types of documents.
- **-v:** This is an option or flag that you pass to the Vim command. In this context, it's used to tell Vim to display its version information.
- **Write Prometheus File**

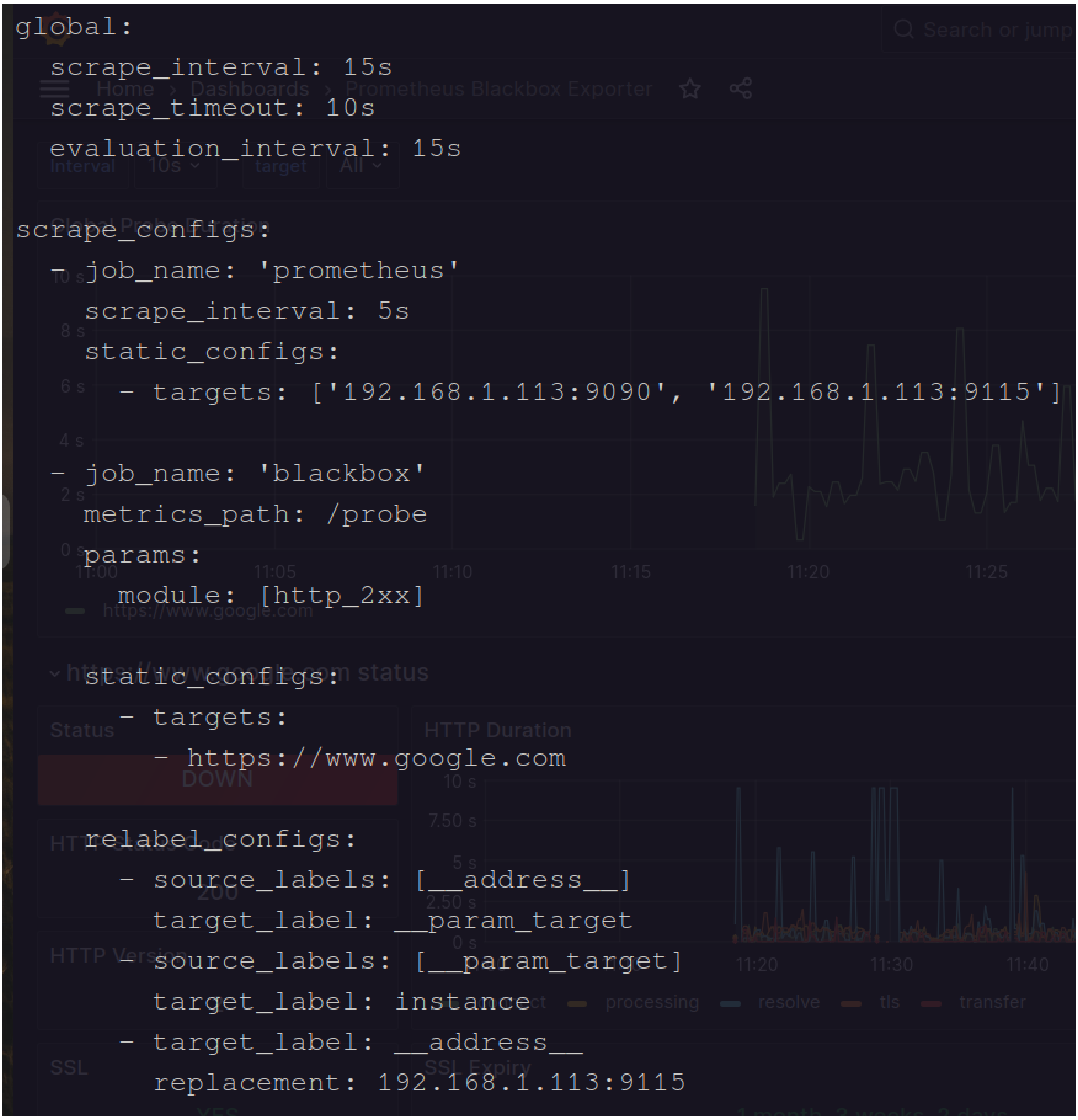
  vim prometheus.yml

When you open a **prometheus.yml** file in Vim, you need to first press **"i" (insert**) to start editing. Then you should enter the following code. After making the necessary changes according to your coding, you should press **"Esc" (escape)** and then type **":wq!" (write and quit forcefully)** to save the file and exit it. This will save your file and close it.

Paste code in this configuration file .

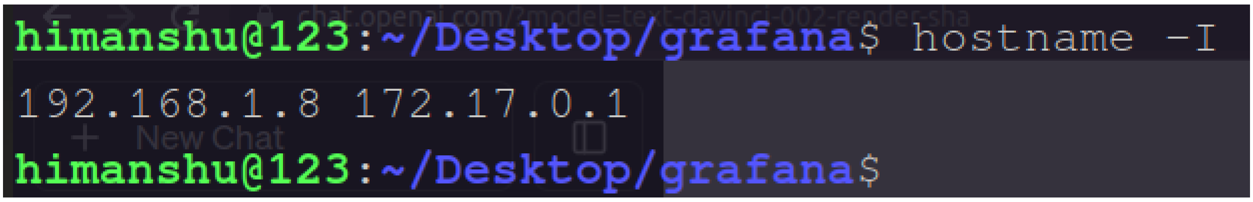
```
global:
  scrape_interval: 15s
  scrape_timeout: 10s
  evaluation_interval: 15s

scrape_configs:
  - job_name: 'prometheus'
    scrape_interval: 5s
    static_configs:
      - targets: ['192.168.1.113:9090', '192.168.1.113:9115']

  - job_name: 'blackbox'
```

```
    metrics_path: /probe
    params:
      module: [http_2xx]

    static_configs:
      - targets:
        - https://www.google.com

    relabel_configs:
      - source_labels: [__address__]
        target_label: __param_target
      - source_labels: [__param_target]
        target_label: instance
      - target_label: __address__
        replacement: 192.168.1.113:9115
```



```
global:
  scrape_interval: 15s
  scrape_timeout: 10s
  evaluation_interval: 15s

scrape_configs:
  - job_name: 'prometheus'
    scrape_interval: 5s
    static_configs:
      - targets: ['192.168.1.113:9090', '192.168.1.113:9115']

  - job_name: 'blackbox'
    metrics_path: /probe
    params:
      module: [http_2xx]

    static_configs:
      - targets:
        - https://www.google.com

    relabel_configs:
      - source_labels: [__address__]
        target_label: __param_target
      - source_labels: [__param_target]
        target_label: instance
      - target_label: __address__
        replacement: 192.168.1.113:9115
```

```
hostname -I
```



```
himanshu@123:~/Desktop/grafana$ hostname -I
192.168.1.8 172.17.0.1
himanshu@123:~/Desktop/grafana$
```

- **starting ip**

You need to add the hostname and IP address in your Prometheus.yml file, and you can choose any URL for monitoring. I have monitored on **Google's website**. You can choose any URL according to your preference. https://www.google.com

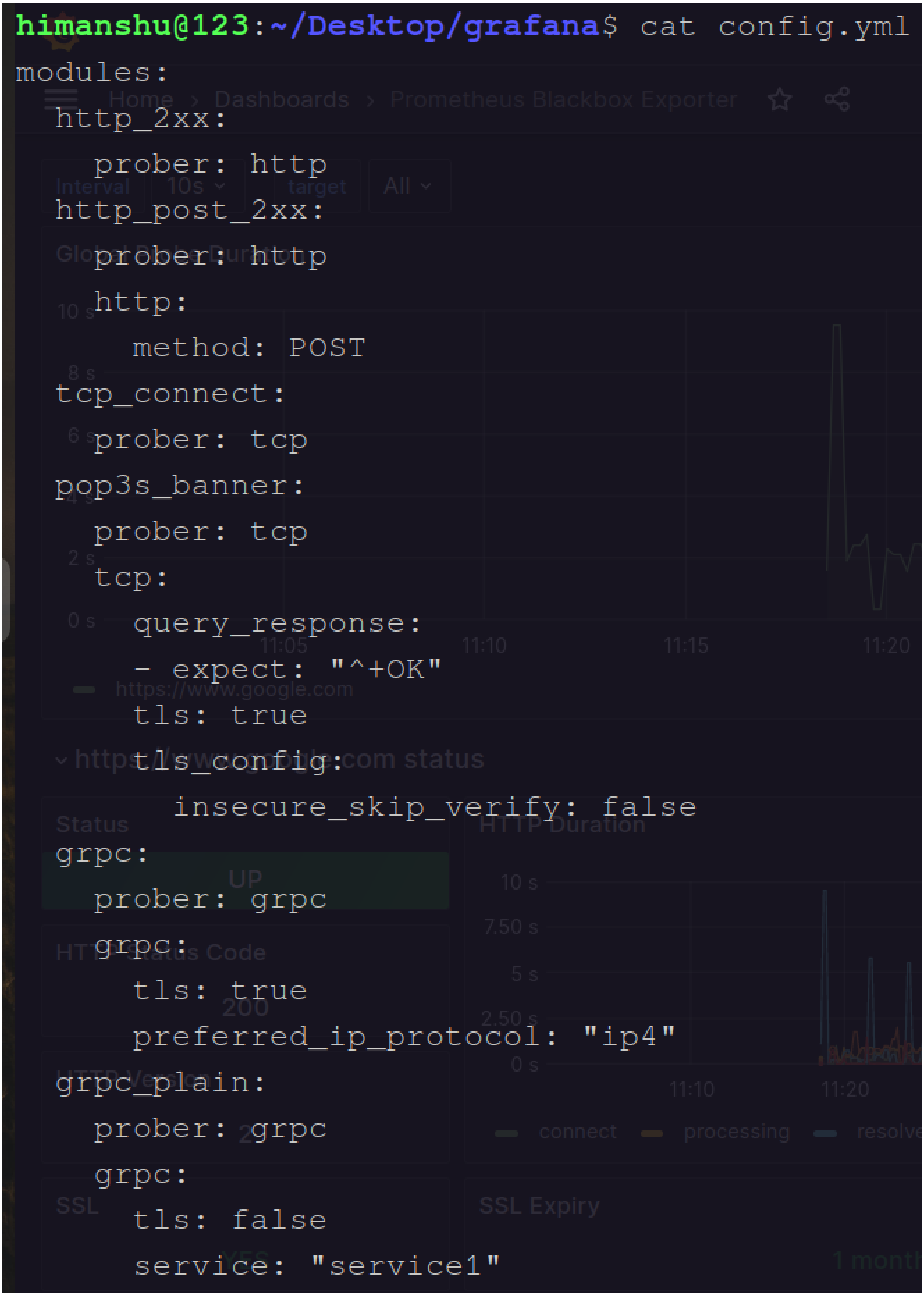
- **Write Configuration File**

  vim config.yml

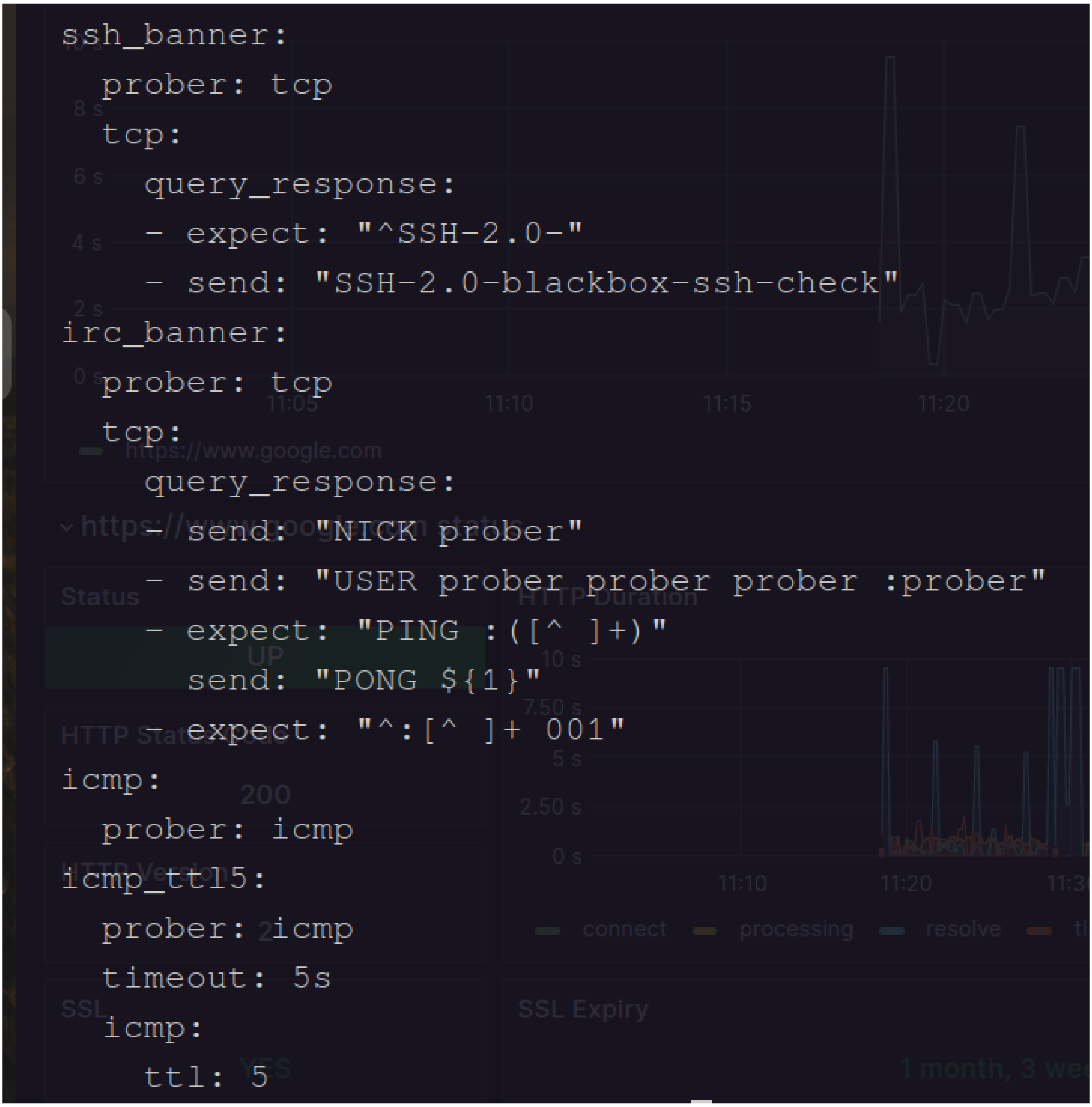
When you open a **config.yml** file in Vim, you need to first press **"i" (insert**) to start editing. Then you should enter the following code. After making the necessary changes according to your coding, you should press **"Esc" (escape)** and then type **":wq!" (write and quit forcefully)** to save the file and exit it. This will save your file and close it.

Paste code in this configuration file .

```
modules:
  http_2xx:
    prober: http
  http_post_2xx:
    prober: http
    http:
      method: POST
  tcp_connect:
    prober: tcp
  pop3s_banner:
    prober: tcp
    tcp:
      query_response:
      - expect: "^+OK"
      tls: true
      tls_config:
        insecure_skip_verify: false
  grpc:
    prober: grpc
    grpc:
      tls: true
      preferred_ip_protocol: "ip4"
  grpc_plain:
    prober: grpc
    grpc:
      tls: false
      service: "service1"
  ssh_banner:
    prober: tcp
    tcp:
      query_response:
      - expect: "^SSH-2.0-"
      - send: "SSH-2.0-blackbox-ssh-check"
  irc_banner:
    prober: tcp
    tcp:
      query_response:
      - send: "NICK prober"
      - send: "USER prober prober prober :prober"
      - expect: "PING :([^ ]+)"
        send: "PONG ${1}"
      - expect: "^:[^ ]+ 001"
  icmp:
    prober: icmp
  icmp_ttl5:
    prober: icmp
    timeout: 5s
    icmp:
      ttl: 5
```



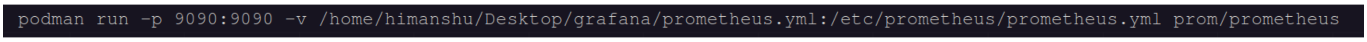
```
himanshu@123:~/Desktop/grafana$ cat config.yml
modules:
  http_2xx:
    prober: http
  http_post_2xx:
    prober: http
    http:
      method: POST
  tcp_connect:
    prober: tcp
  pop3s_banner:
    prober: tcp
    tcp:
      query_response:
      - expect: "^+OK"
      tls: true
      tls_config:
        insecure_skip_verify: false
  grpc:
    prober: grpc
    grpc:
      tls: true
      preferred_ip_protocol: "ip4"
  grpc_plain:
    prober: grpc
    grpc:
      tls: false
      service: "service1"
```



```
ssh_banner:
  prober: tcp
  tcp:
    query_response:
    - expect: "^SSH-2.0-"
    - send: "SSH-2.0-blackbox-ssh-check"
irc_banner:
  prober: tcp
  tcp:
    query_response:
    - send: "NICK prober"
    - send: "USER prober prober prober :prober"
    - expect: "PING :([^ ]+)"
      send: "PONG ${1}"
    - expect: "^:[^ ]+ 001"
icmp:
  prober: icmp
icmp_ttl5:
  prober: icmp
  timeout: 5s
  icmp:
    ttl: 5
```

Step 4 . Run the container Prometheus on podman .

```
podman run -d -p 9090:9090 -v
/home/himanshu/Desktop/grafana/prometheus.yml:/etc/prometheus/prometheus.ym
l --name prometheus-container prom/prometheus
```



```
podman run -p 9090:9090 -v /home/himanshu/Desktop/grafana/prometheus.yml:/etc/prometheus/prometheus.yml prom/prometheus
```

- **podman run:** This is the command to run a container using Podman, which is an alternative to Docker for managing containers.
- **-d:** This flag indicates that you want to run the container in detached mode, meaning it will run in the background.

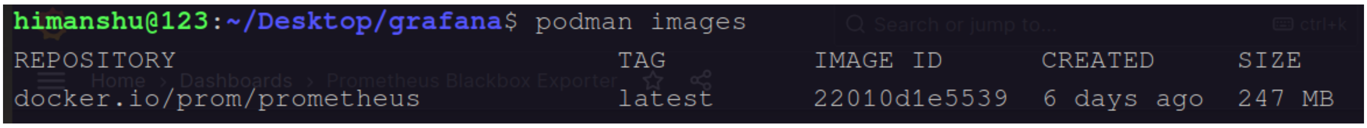
- **-p 9090:9090:** Which one is the host port and which one is the container port? - In the command -p 9090:9090, the **first** 9090 represents the host port, and the **second** 9090 represents the container port. This means that traffic sent to port 9090 on the host will be forwarded to port 9090 inside the container.

- **-v /home/himanshu/Desktop/grafana/prometheus.yml:/etc/prometheus/prometheus.yml:** This flag is used to create a volume mount. It maps the local file /home/himanshu/Desktop/grafana/prometheus.yml to the container's path /etc/prometheus/prometheus.yml. This is a common practice for providing configuration files to a container.

- **--name prometheus-container:** This flag assigns a name to the container. In this case, the container is named "prometheus-container."

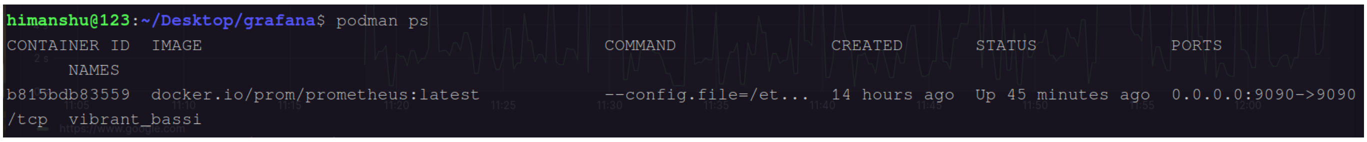
- **prom/prometheus:** This is the Docker image you want to run. It appears to be the official Prometheus Docker image, which is used to run the Prometheus monitoring system.

- `podman images` : check images .



```
himanshu@123:~/Desktop/grafana$ podman images
REPOSITORY                  TAG       IMAGE ID      CREATED     SIZE
docker.io/prom/prometheus   latest    22010d1e5539  6 days ago  247 MB
```

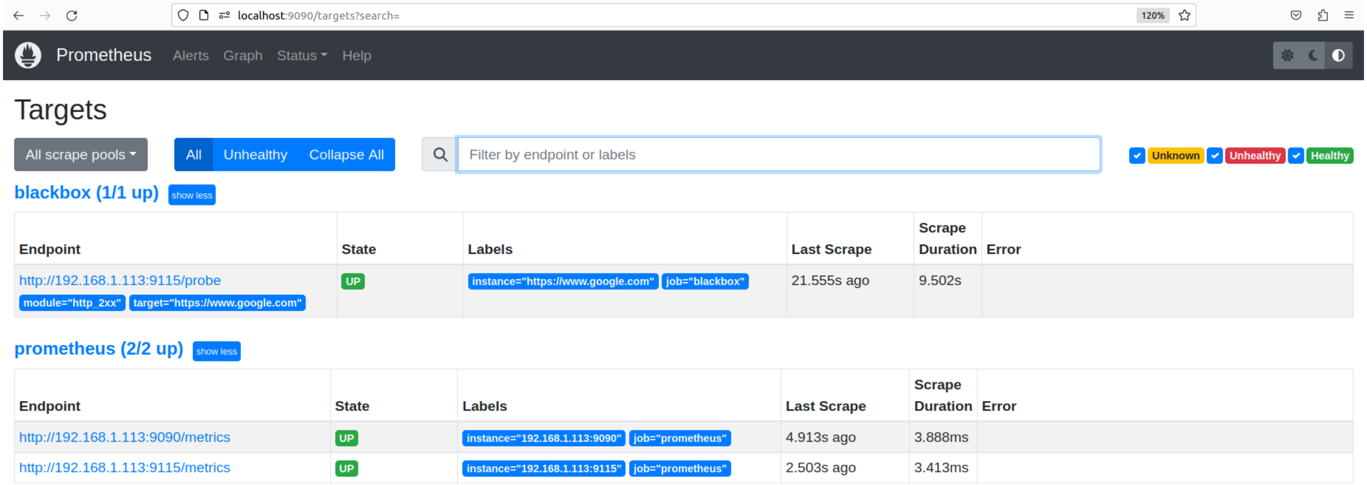
- `podman ps` : check container .



```
himanshu@123:~/Desktop/grafana$ podman ps
CONTAINER ID  IMAGE                                 COMMAND               CREATED       STATUS             PORTS
    NAMES
b815bdb83559  docker.io/prom/prometheus:latest      --config.file=/et...  14 hours ago  Up 45 minutes ago  0.0.0.0:9090->9090
/tcp  vibrant_bassi
```

- show on Localhost : http://localhost:9090/targets?search=

In the context of Prometheus, **"targets"** typically refer to the services or endpoints that Prometheus scrapes data from. The URL **"localhost:9090"** is often used to access the Prometheus web interface. It's possible that the targets you are trying to access are not configured properly in your Prometheus instance.



localhost:9090/targets?search=
120%
Prometheus
Alerts
Graph
Status
Help
All scrape pools
All
Unhealthy
Collapse All
Filter by endpoint or labels
☑ Unknown ☑ Unhealthy ☑ Healthy
show less

Targets

blackbox (1/1 up)

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://192.168.1.113:9115/probe module="http_2xx" target="https://www.google.com"	UP	instance="https://www.google.com" job="blackbox"	21.555s ago	9.502s	

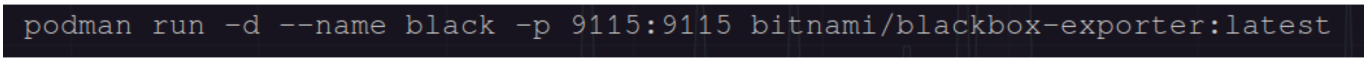
prometheus (2/2 up)

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://192.168.1.113:9090/metrics	UP	instance="192.168.1.113:9090" job="prometheus"	4.913s ago	3.888ms	
http://192.168.1.113:9115/metrics	UP	instance="192.168.1.113:9115" job="prometheus"	2.503s ago	3.413ms	

If your container is **down**, the first thing you need to do is run the commands for Blackbox Exporter and Grafana, which will bring your container back **up**.

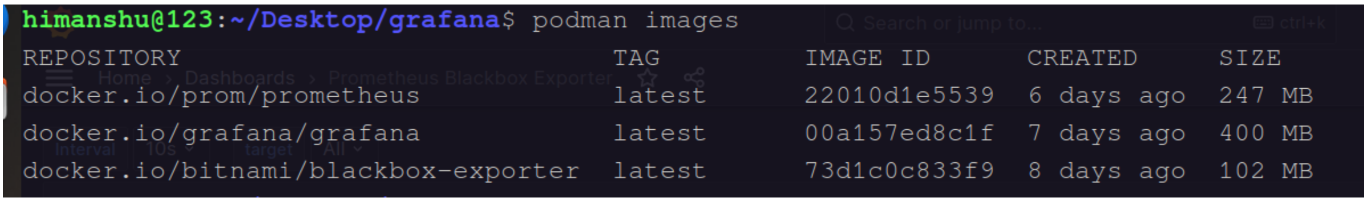
Step 5 . Run the container Blackbox Exporter on podman .

```
podman run -d --name black -p 9115:9115  bitnami/blackbox-exporter:latest
```



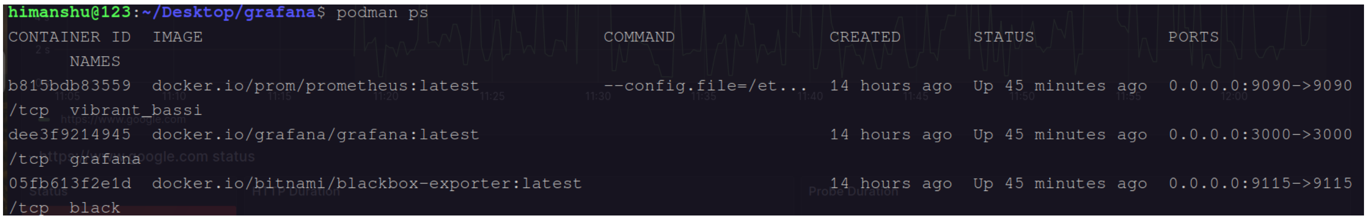
```
podman run -d --name black -p 9115:9115 bitnami/blackbox-exporter:latest
```

- **podman run:** This is the basic command for running a container with Podman.
- **-d:** This flag stands for "detached" mode, which means that the container will run in the background, and you'll get your terminal prompt back immediately.
- **--name black:** This flag assigns a name "black" to the running container, allowing you to easily reference it by name instead of a container ID.
- **-p 9115:9115:** This flag maps ports between the host and the container. In this case, it's mapping port 9115 on the host to port 9115 in the container. This is useful if the container is running a service that you want to access from your host machine.
- **bitnami/blackbox-exporter:latest:** This is the name of the Docker image that you want to run as a container. It's specifying the image "bitnami/blackbox-exporter" with the "latest" tag, meaning the latest available version of that image.
- `podman images` : check images .



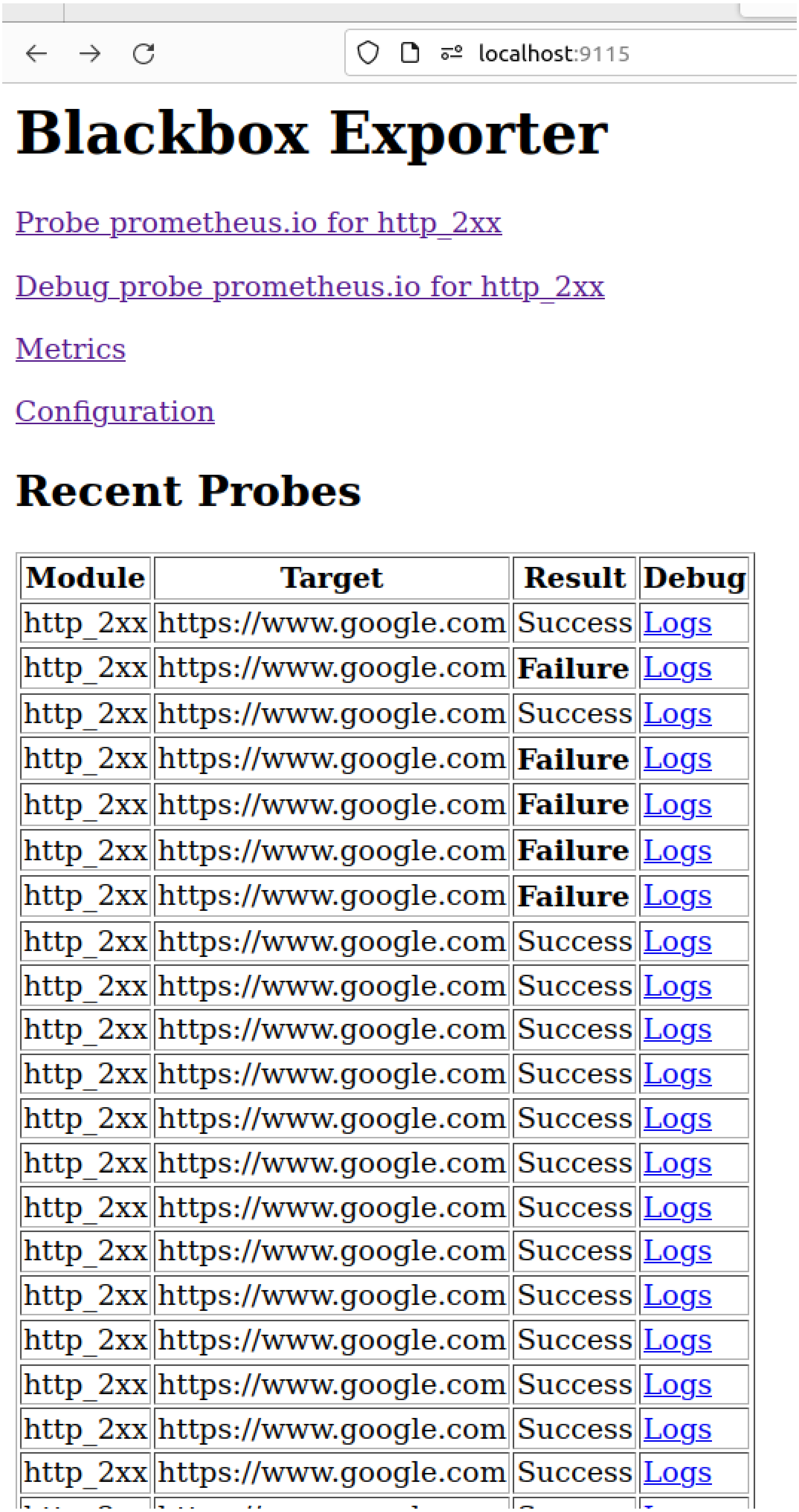
```
himanshu@123:~/Desktop/grafana$ podman images
REPOSITORY                          TAG      IMAGE ID      CREATED     SIZE
docker.io/prom/prometheus           latest   22010d1e5539  6 days ago  247 MB
docker.io/grafana/grafana           latest   00a157ed8c1f  7 days ago  400 MB
docker.io/bitnami/blackbox-exporter latest   73d1c0c833f9  8 days ago  102 MB
```

- `podman ps` : check container .



```
himanshu@123:~/Desktop/grafana$ podman ps
CONTAINER ID  IMAGE                                      COMMAND               CREATED       STATUS               PORTS
     NAMES
b815bdb83559  docker.io/prom/prometheus:latest           --config.file=/et...  14 hours ago  Up 45 minutes ago  0.0.0.0:9090->9090
/tcp  vibrant_bassi
dee3f9214945  docker.io/grafana/grafana:latest                                 14 hours ago  Up 45 minutes ago  0.0.0.0:3000->3000
/tcp  grafana
05fb613f2e1d  docker.io/bitnami/blackbox-exporter:latest                       14 hours ago  Up 45 minutes ago  0.0.0.0:9115->9115
/tcp  black
```

- show on Localhost : http://localhost:9115



Blackbox Exporter

Probe prometheus.io for http_2xx

Debug probe prometheus.io for http_2xx

Metrics

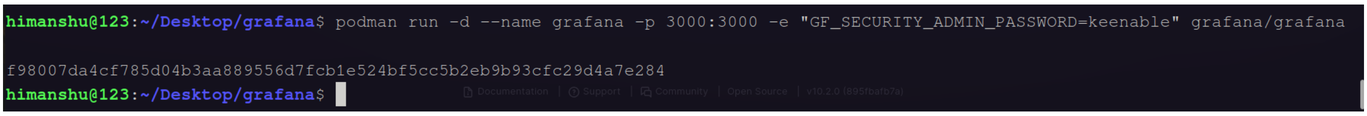
Configuration

Recent Probes

Module	Target	Result	Debug
http_2xx	https://www.google.com	Success	Logs
http_2xx	https://www.google.com	**Failure**	Logs
http_2xx	https://www.google.com	Success	Logs
http_2xx	https://www.google.com	**Failure**	Logs
http_2xx	https://www.google.com	**Failure**	Logs
http_2xx	https://www.google.com	**Failure**	Logs
http_2xx	https://www.google.com	**Failure**	Logs
http_2xx	https://www.google.com	Success	Logs
http_2xx	https://www.google.com	Success	Logs
http_2xx	https://www.google.com	Success	Logs
http_2xx	https://www.google.com	Success	Logs
http_2xx	https://www.google.com	Success	Logs
http_2xx	https://www.google.com	Success	Logs
http_2xx	https://www.google.com	Success	Logs
http_2xx	https://www.google.com	Success	Logs
http_2xx	https://www.google.com	Success	Logs
http_2xx	https://www.google.com	Success	Logs
http_2xx	https://www.google.com	Success	Logs
http_2xx	https://www.google.com	Success	Logs
http_2xx	https://www.google.com	Success	Logs

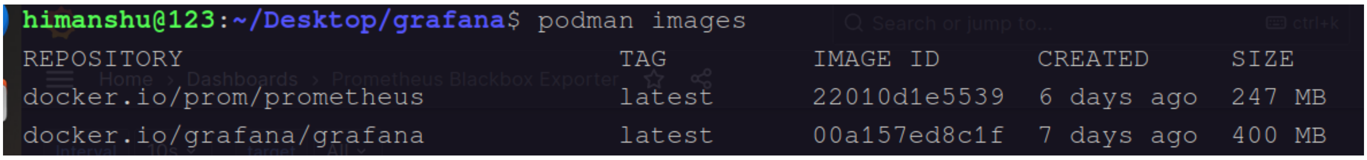
Step 6 . Run the container Grafana on podman .

```
podman run -d --name grafana -p 3000:3000 -e
"GF_SECURITY_ADMIN_PASSWORD=keenable" grafana/grafana
```



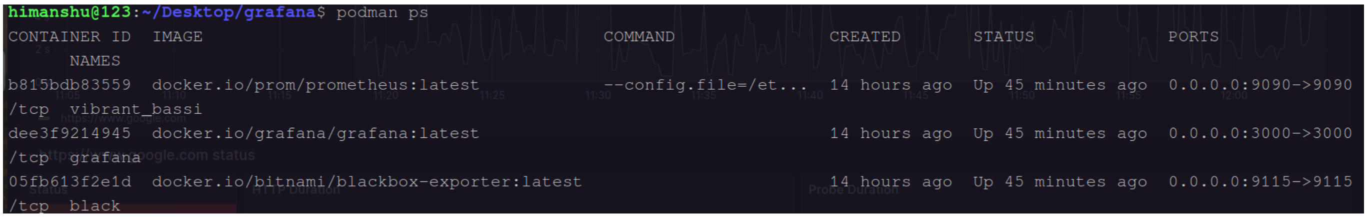
```
himanshu@123:~/Desktop/grafana$ podman run -d --name grafana -p 3000:3000 -e "GF_SECURITY_ADMIN_PASSWORD=keenable" grafana/grafana

f98007da4cf785d04b3aa889556d7fcb1e524bf5cc5b2eb9b93cfc29d4a7e284
himanshu@123:~/Desktop/grafana$
```

- **podman run:** This is the command to run a container using Podman, a containerization tool similar to Docker.
- **-d:** This flag stands for "detached" mode, which means that the container will run in the background as a daemon.
- **--name grafana:** This flag assigns a name to the container, in this case, "grafana." This name can be used to reference and manage the container.
- **-p 3000:3000**: This flag is used to map ports between the host machine and the container. It specifies that port 3000 on the host should be mapped to port 3000 inside the Grafana container. This is important for accessing Grafana's web interface, as the Grafana server runs on port 3000 by default.
- **-e "GF_SECURITY_ADMIN_PASSWORD=keenable":** This flag is used to set an environment variable within the container. In this case, it's setting the Grafana admin user's password to "admin." This is a common initial setup step to secure your Grafana instance.
- **grafana/grafana:** This is the name of the Docker image that you want to run. It specifies that you want to run the official Grafana Docker image from the "grafana" repository on Docker Hub.
- `podman images` : check images .



```
himanshu@123:~/Desktop/grafana$ podman images
REPOSITORY                    TAG         IMAGE ID      CREATED      SIZE
docker.io/prom/prometheus     latest      22010d1e5539  6 days ago   247 MB
docker.io/grafana/grafana     latest      00a157ed8c1f  7 days ago   400 MB
```

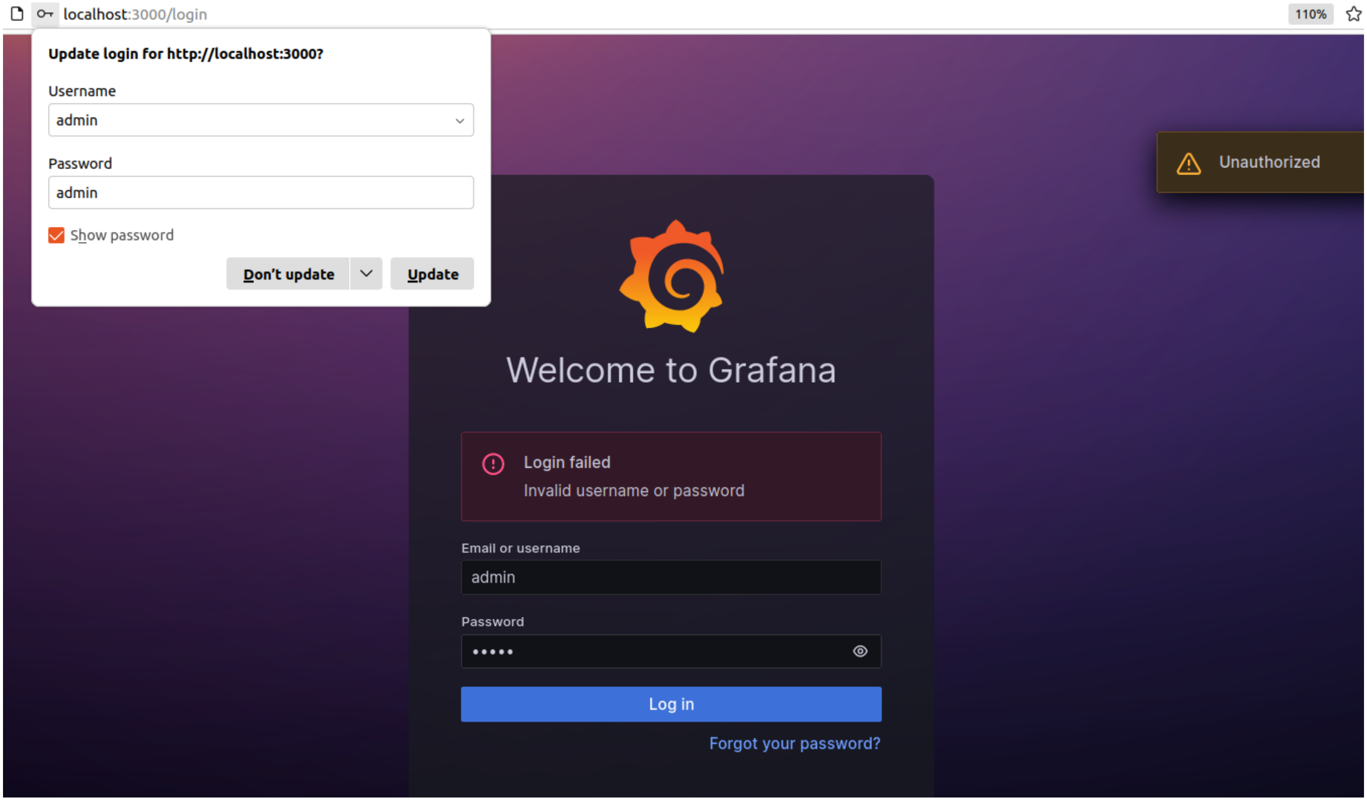
- `podman restart <container id>`: **Restart all container**
- `podman ps` : check container .



```
himanshu@123:~/Desktop/grafana$ podman ps
CONTAINER ID  IMAGE                                    COMMAND               CREATED       STATUS                 PORTS
    NAMES
b815bdb83559  docker.io/prom/prometheus:latest         --config.file=/et...  14 hours ago  Up 45 minutes ago  0.0.0.0:9090->9090
/tcp  vibrant_bassi
dee3f9214945  docker.io/grafana/grafana:latest                               14 hours ago  Up 45 minutes ago  0.0.0.0:3000->3000
/tcp  grafana
05fb613f2e1d  docker.io/bitnami/blackbox-exporter:latest                     14 hours ago  Up 45 minutes ago  0.0.0.0:9115->9115
/tcp  black
```

- show on Localhost : http://localhost:3000

If we enter the ID and password as **'admin admin'**, Grafana will not open.



localhost:3000/login

Update login for http://localhost:3000?

Username: admin

Password: admin

☑ Show password

Don't update | Update

Unauthorized

Welcome to Grafana

Login failed
Invalid username or password

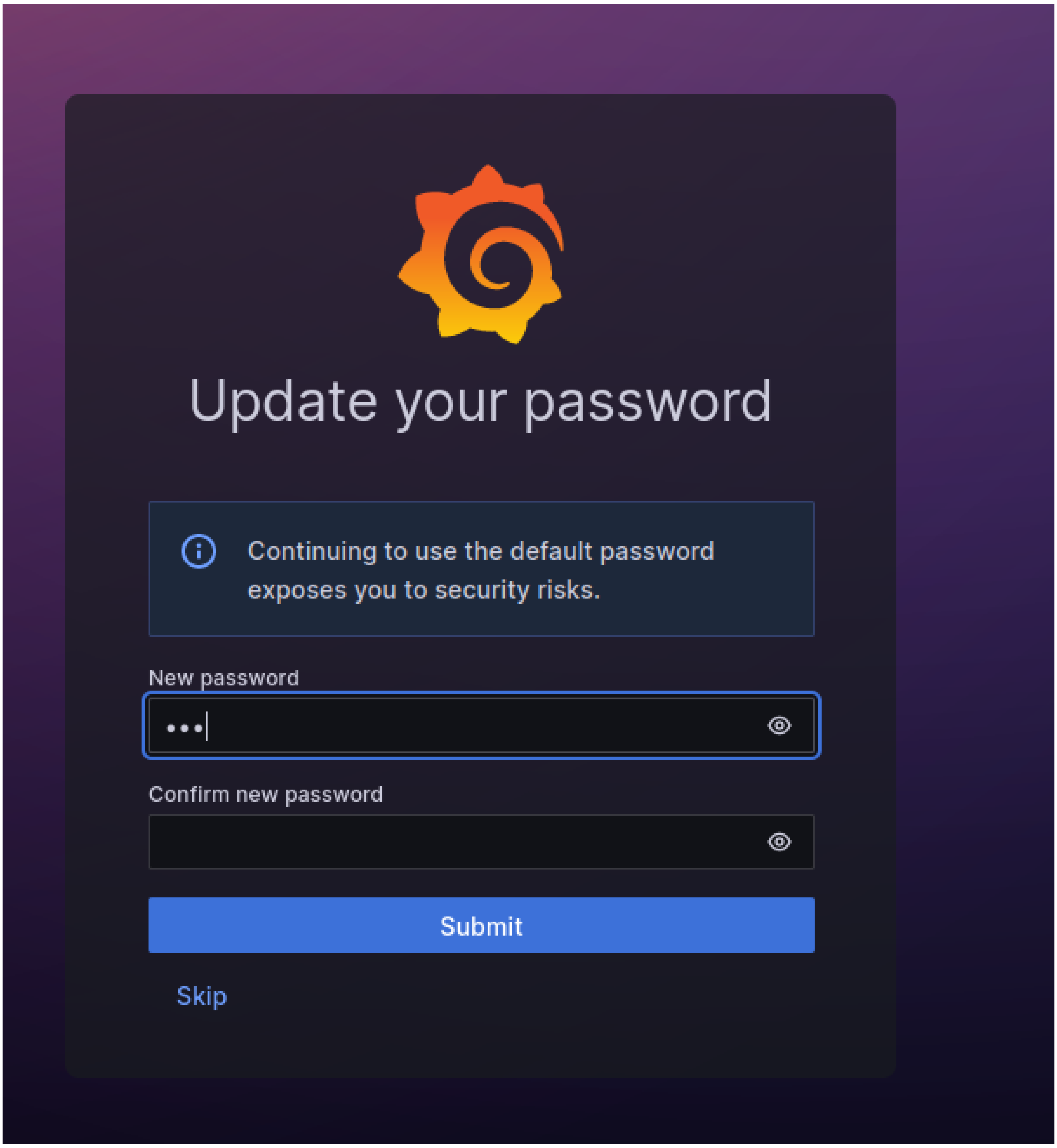
Email or username: admin

Password: •••••

Log in

Forgot your password?

- As soon as I entered the **username 'admin' and the password 'keenable,'** my Grafana opened.

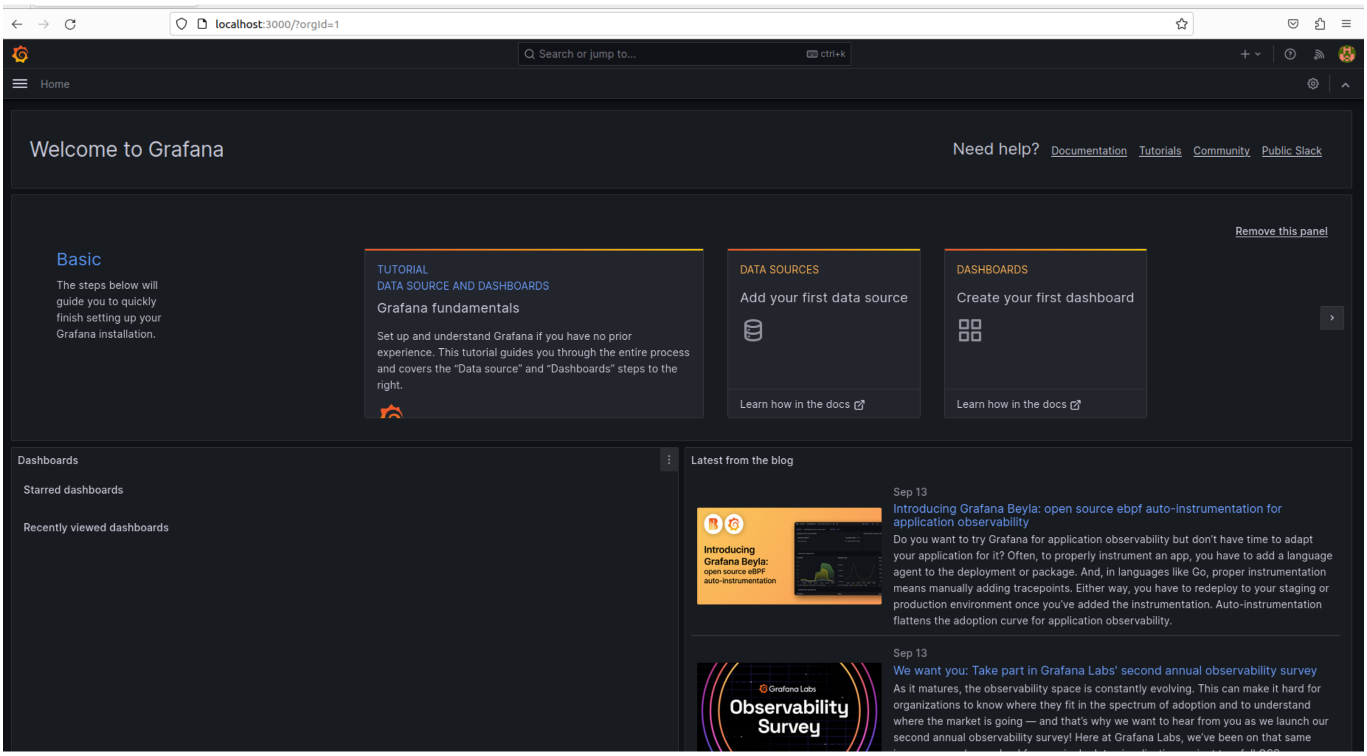


Update your password

Continuing to use the default password exposes you to security risks.

New password: •••

Confirm new password:



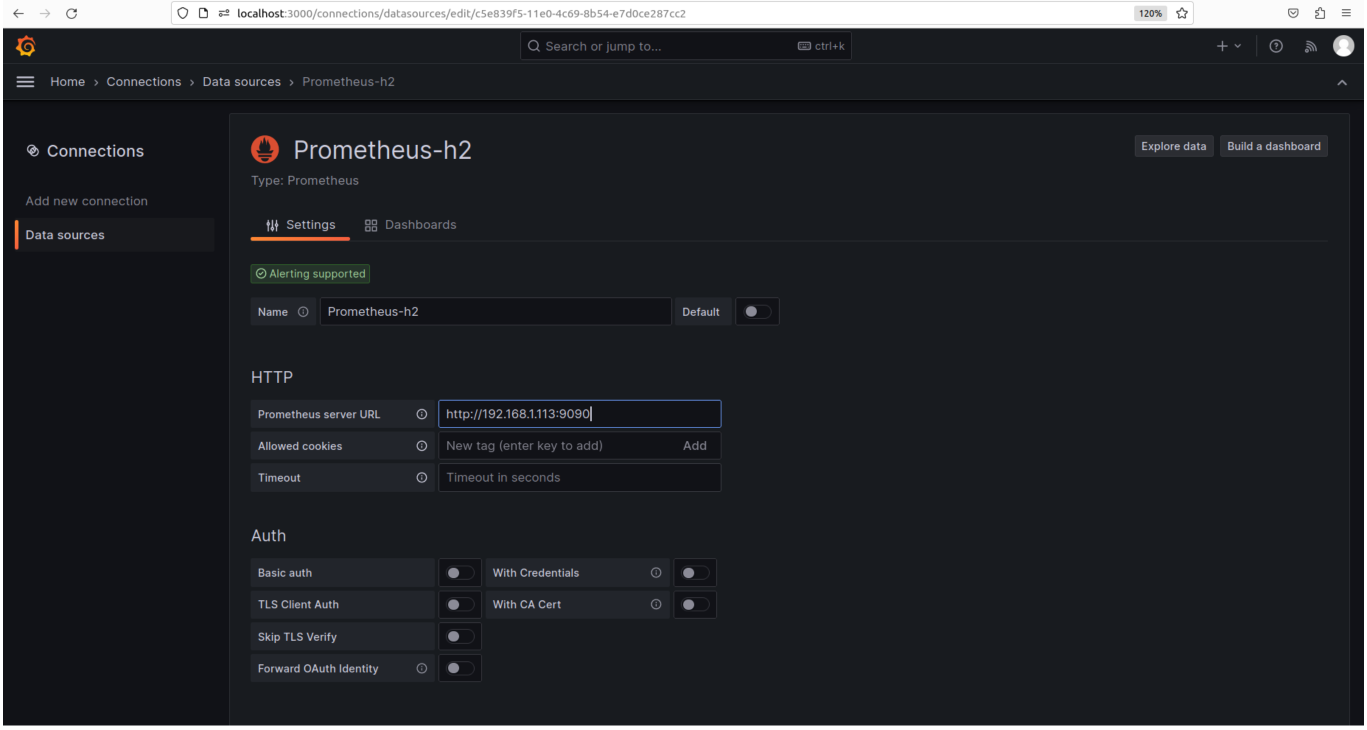
The steps below will guide you to quickly finish setting up your Grafana installation.

Set up and understand Grafana if you have no prior experience. This tutorial guides you through the entire process and covers the "Data source" and "Dashboards" steps to the right.

Sep 13
Introducing Grafana Beyla: open source ebpf auto-instrumentation for application observability
Do you want to try Grafana for application observability but don't have time to adapt your application for it? Often, to properly instrument an app, you have to add a language agent to the deployment or package. And, in languages like Go, proper instrumentation means manually adding tracepoints. Either way, you have to redeploy to your staging or production environment once you've added the instrumentation. Auto-instrumentation flattens the adoption curve for application observability.

Sep 13
We want you: Take part in Grafana Labs' second annual observability survey
As it matures, the observability space is constantly evolving. This can make it hard for organizations to know where they fit in the spectrum of adoption and to understand where the market is going — and that's why we want to hear from you as we launch our second annual observability survey! Here at Grafana Labs, we've been on that same

After opening Grafana, the first thing you need to do is add a data source for Prometheus.



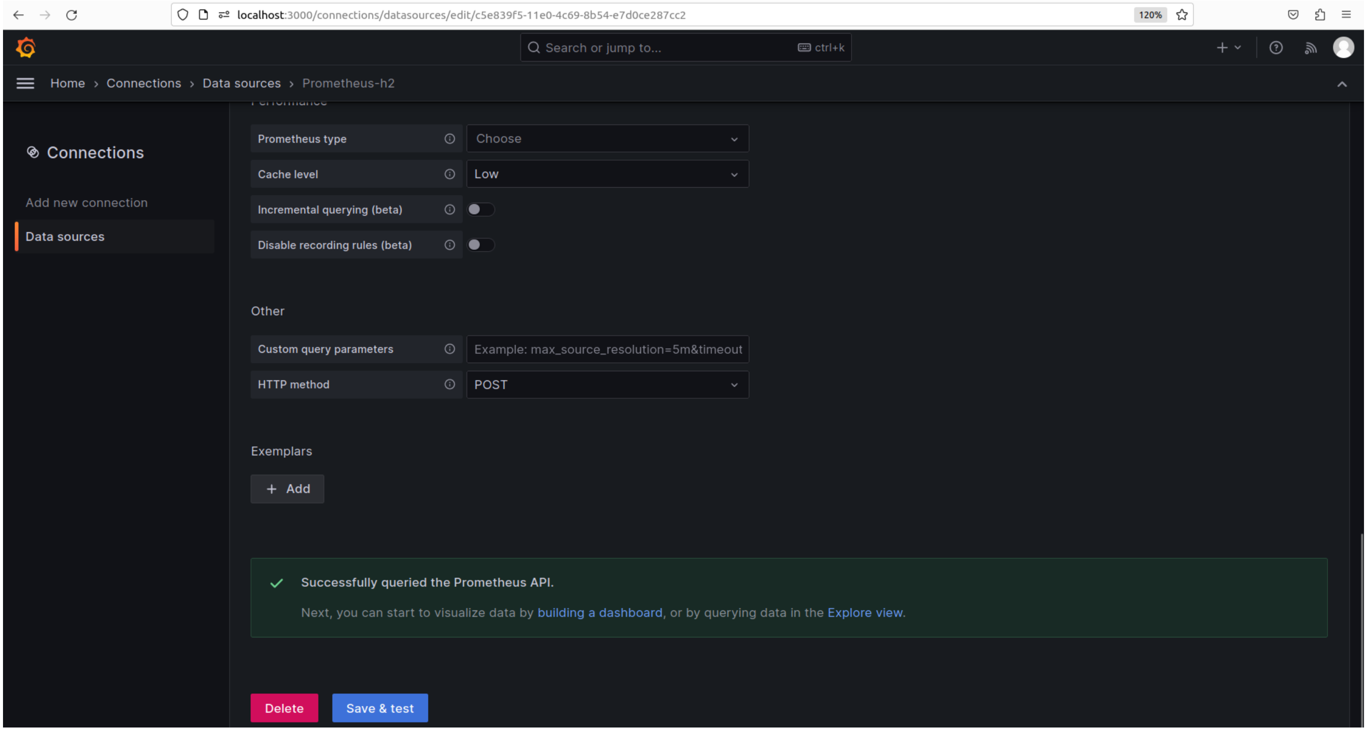
Prometheus-h2
Type: Prometheus

Name: Prometheus-h2 Default: off

HTTP
Prometheus server URL: http://192.168.1.113:9090
Allowed cookies: New tag (enter key to add) Add
Timeout: Timeout in seconds

Auth
Basic auth: off With Credentials: off
TLS Client Auth: off With CA Cert: off
Skip TLS Verify: off
Forward OAuth Identity: off

In order to configure Prometheus, you will need to add the following **URL: http://Hostname starting IP:9090**, which should include the **port and hostname** of your Prometheus.



localhost:3000/connections/datasources/edit/c5e839f5-11e0-4c69-8b54-e7d0ce287cc2
Search or jump to...
ctrl+k
Home › Connections › Data sources › Prometheus-h2
Connections
Add new connection
Data sources
Delete
Save & test

Prometheus type: Choose

Cache level: Low

Incremental querying (beta)

Disable recording rules (beta)

Other

Custom query parameters: Example: max_source_resolution=5m&timeout

HTTP method: POST

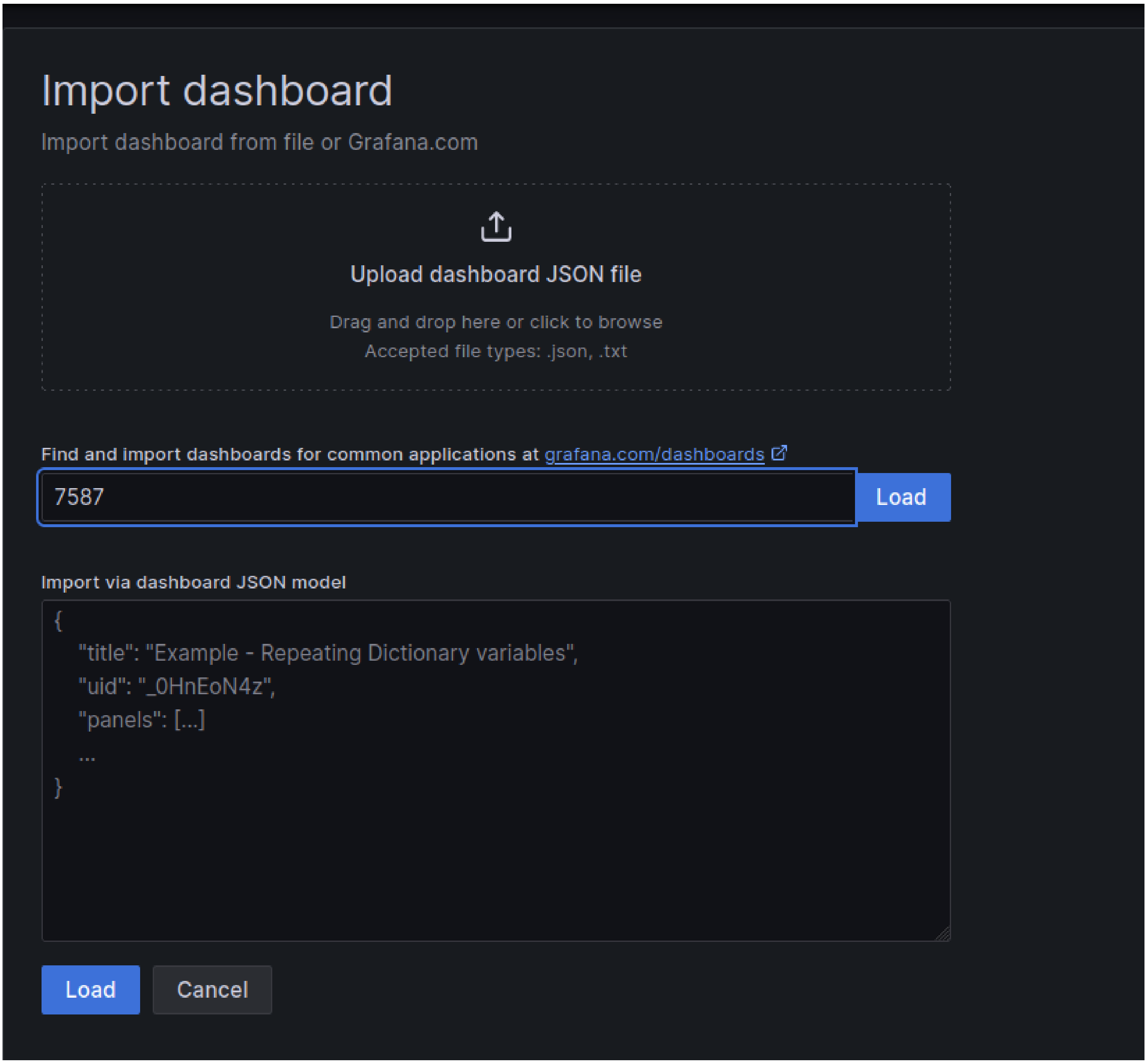
Exemplars

+ Add

Successfully queried the Prometheus API.

Next, you can start to visualize data by building a dashboard, or by querying data in the Explore view.

After adding the data source, you will need to **import** a dashboard. To do that, you need to enter the **Grafana dashboard IP as 7587**. When you enter the dashboard IP and add the Prometheus data source, your dashboard will open.



Import dashboard

Import dashboard from file or Grafana.com

Upload dashboard JSON file

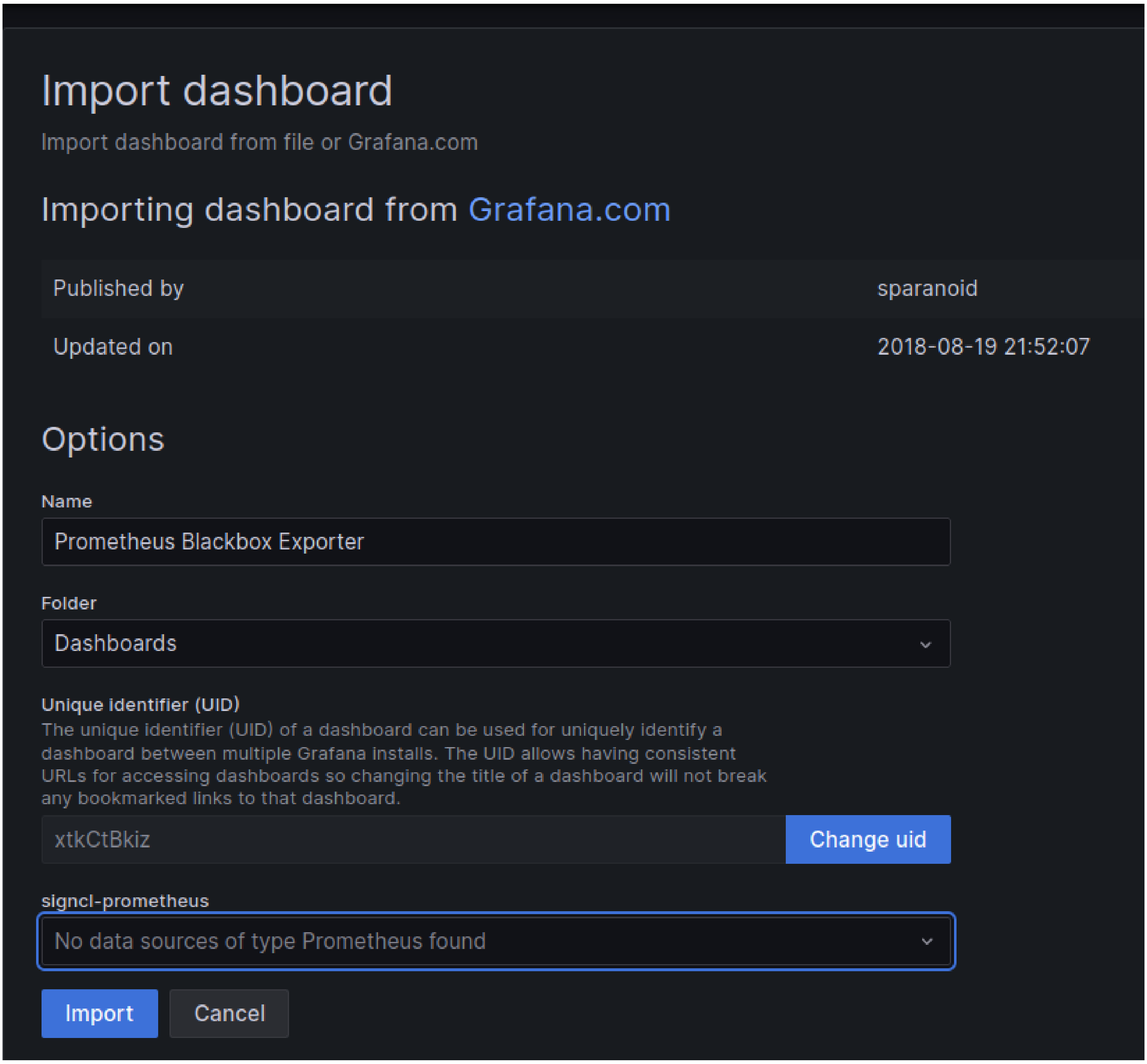
Drag and drop here or click to browse
Accepted file types: .json, .txt

Find and import dashboards for common applications at grafana.com/dashboards

7587

Import via dashboard JSON model

```
{
    "title": "Example - Repeating Dictionary variables",
    "uid": "_0HnEoN4z",
    "panels": [...]
    ...
}
```



Import dashboard

Import dashboard from file or Grafana.com

Importing dashboard from Grafana.com

Published by	sparanoid
Updated on	2018-08-19 21:52:07

Options

Name

Prometheus Blackbox Exporter

Folder

Dashboards

Unique identifier (UID)

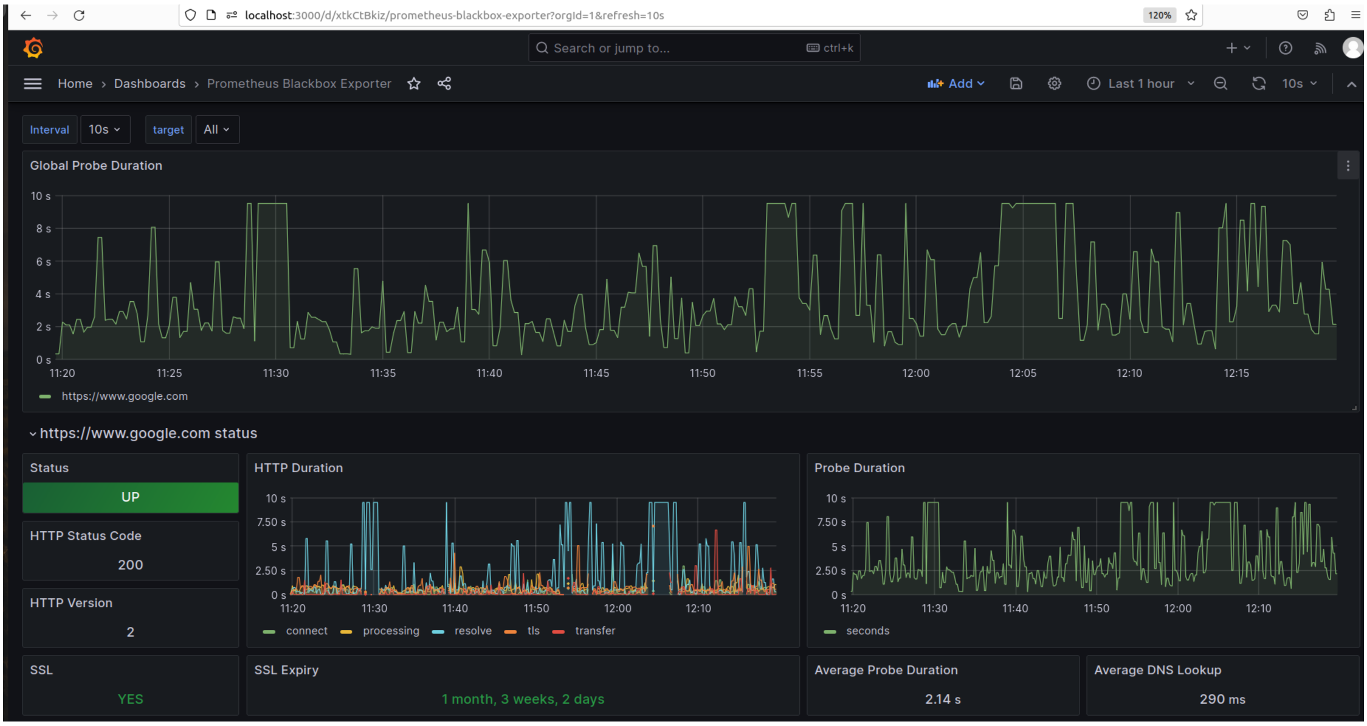
The unique identifier (UID) of a dashboard can be used for uniquely identify a dashboard between multiple Grafana installs. The UID allows having consistent URLs for accessing dashboards so changing the title of a dashboard will not break any bookmarked links to that dashboard.

xtkCtBkiz

signcl-prometheus

No data sources of type Prometheus found

As soon as you import it, your Grafana dashboard will open.



localhost:3000/d/xtkCtBkiz/prometheus-blackbox-exporter?orgId=1&refresh=10s

Home › Dashboards › Prometheus Blackbox Exporter

Interval 10s target All

Last 1 hour 10s

Global Probe Duration

https://www.google.com status

Status: UP

HTTP Status Code: 200

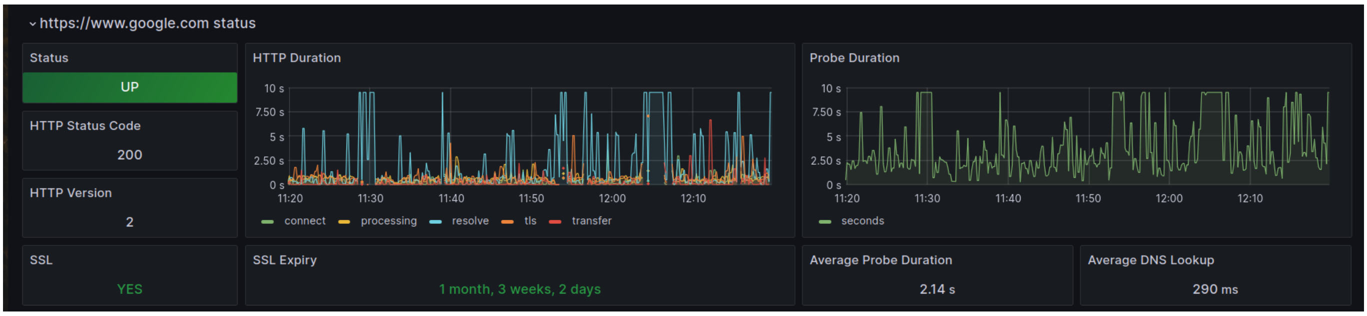
HTTP Version: 2

SSL: YES

SSL Expiry: 1 month, 3 weeks, 2 days

Average Probe Duration: 2.14 s

Average DNS Lookup: 290 ms



https://www.google.com status

Status: UP

HTTP Status Code: 200

HTTP Version: 2

SSL: YES

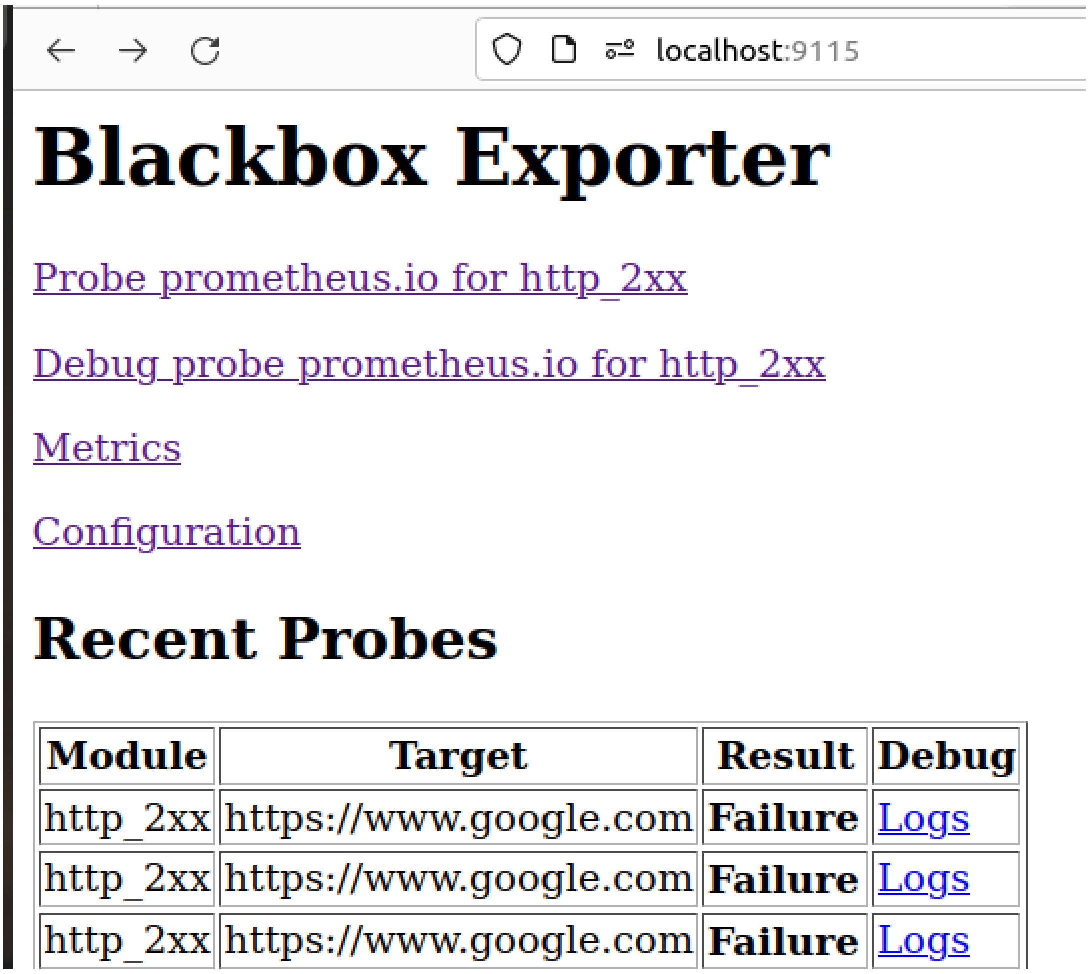
SSL Expiry: 1 month, 3 weeks, 2 days

Average Probe Duration: 2.14 s

Average DNS Lookup: 290 ms

Step 7 . If a website's URL returns a low failure response, then in this situation, the dashboard will be down.

- **Blackbox Exporter**



Blackbox Exporter

Probe prometheus.io for http_2xx

Debug probe prometheus.io for http_2xx

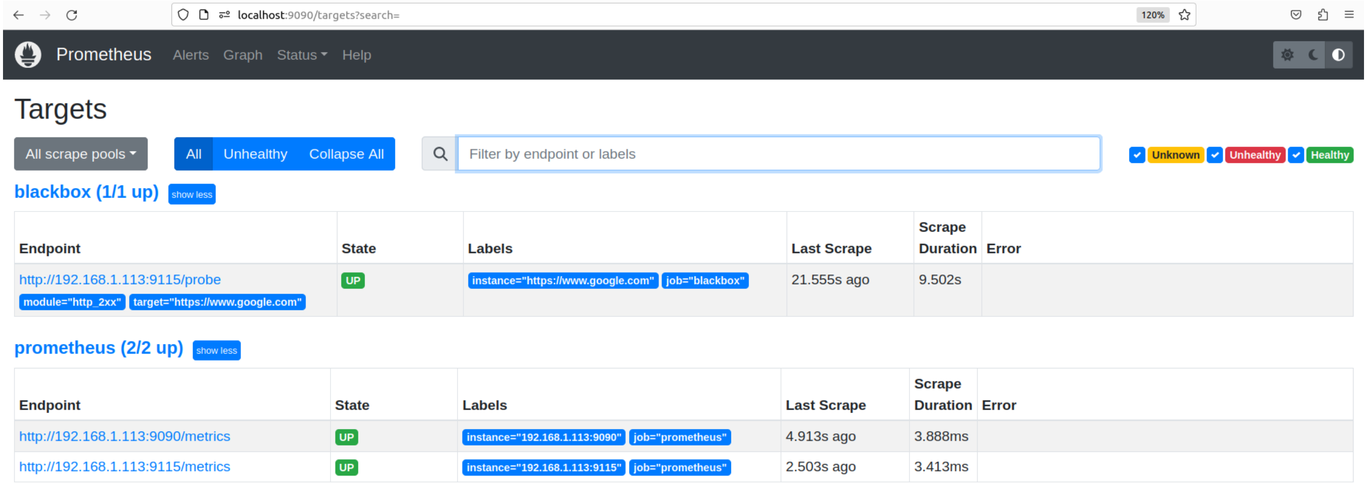
Metrics

Configuration

Recent Probes

Module	Target	Result	Debug
http_2xx	https://www.google.com	Failure	Logs
http_2xx	https://www.google.com	Failure	Logs
http_2xx	https://www.google.com	Failure	Logs

- **Prometheus**



Targets

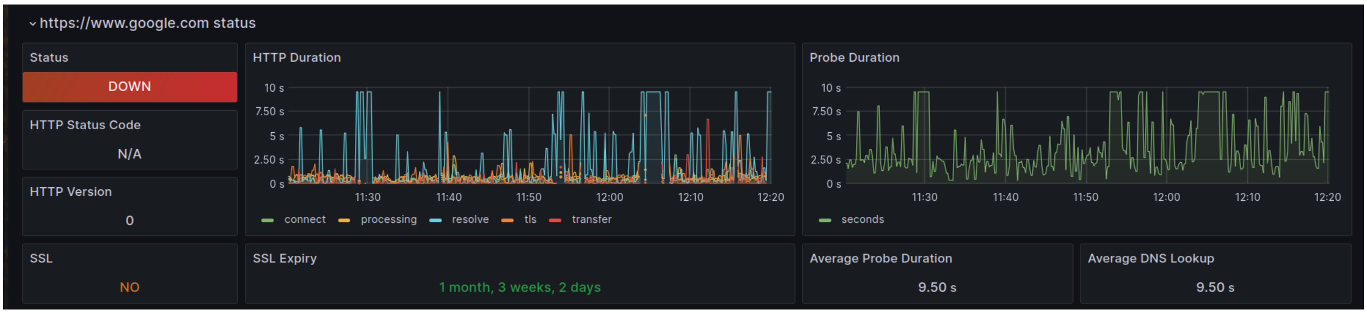
blackbox (1/1 up)

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://192.168.1.113:9115/probe module="http_2xx" target="https://www.google.com"	UP	instance="https://www.google.com" job="blackbox"	21.555s ago	9.502s	

prometheus (2/2 up)

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://192.168.1.113:9090/metrics	UP	instance="192.168.1.113:9090" job="prometheus"	4.913s ago	3.888ms	
http://192.168.1.113:9115/metrics	UP	instance="192.168.1.113:9115" job="prometheus"	2.503s ago	3.413ms	

- **Grafana**



https://www.google.com status

Status: DOWN

HTTP Status Code: N/A

HTTP Version: 0

SSL: NO

SSL Expiry: 1 month, 3 weeks, 2 days

Average Probe Duration: 9.50 s

Average DNS Lookup: 9.50 s