

Grafana

Установка через Docker

```
docker pull grafana/grafana
```

```
root@toorrrp4:~# docker pull grafana/grafana
Using default tag: latest
latest: Pulling from grafana/grafana
7264a8db6415: Pull complete
e9477453234c: Pull complete
80bd18547034: Pull complete
2915f6d75957: Pull complete
8528f10acaa5: Pull complete
5382d38bc0a4: Pull complete
fba2e296b3ec: Pull complete
57c8ca81098f: Pull complete
c9af98a6861f: Pull complete
f114f37263d3: Pull complete
Digest: sha256:29f39e23705d3ef653fa84ca3c01731e0771f1fedbd69ecb99868270cdeb0572
Status: Downloaded newer image for grafana/grafana:latest
docker.io/grafana/grafana:latest
```

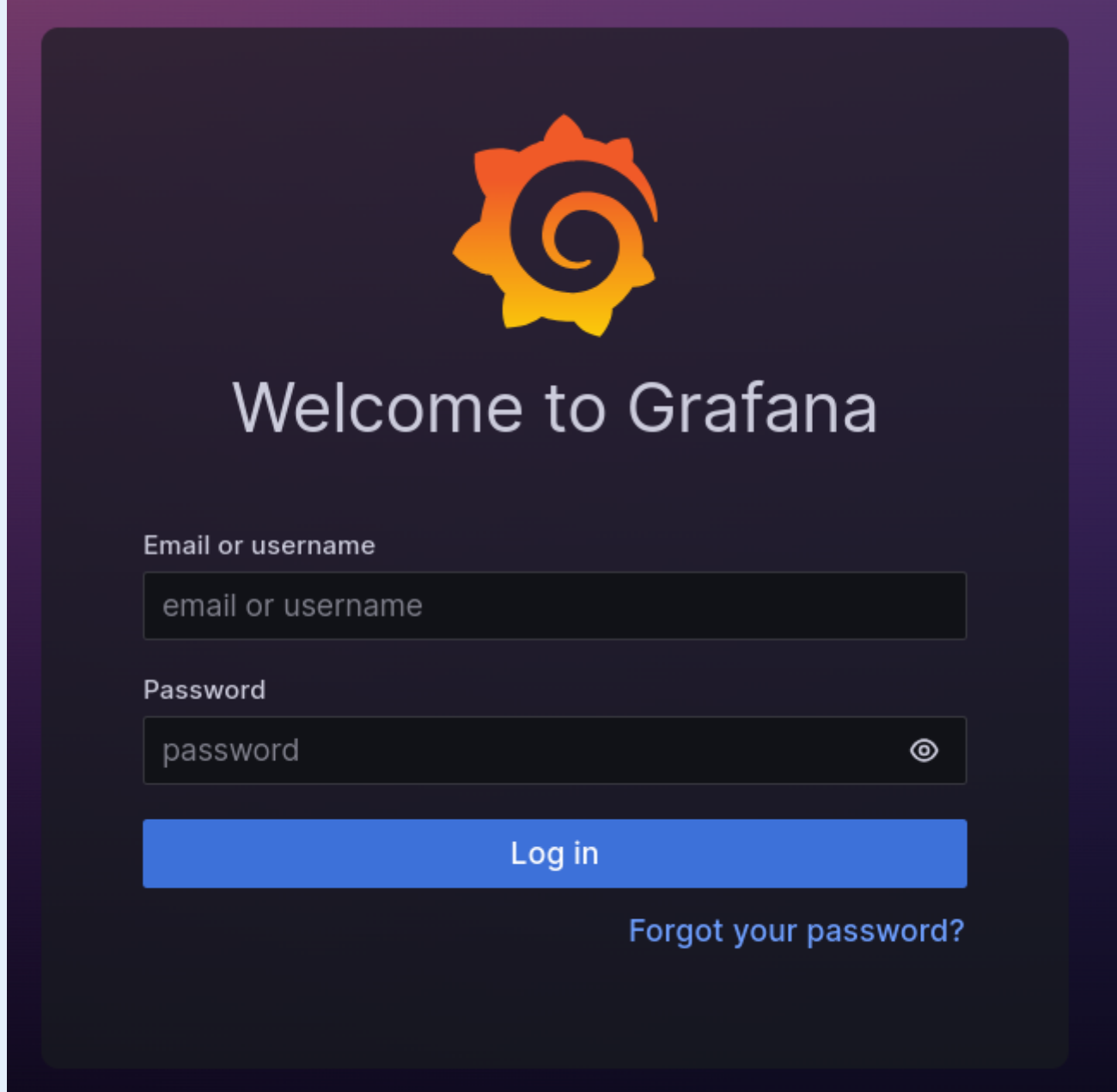
```
docker run -d -p 3000:3000 grafana/grafana
```

```
root@toorrrp4:~# docker run -d -p 3000:3000 grafana/grafana
14979bd5450cd614b9332da720e0cf9e41baf5f91dcc12dc85ae3d3b3ca027b3
root@toorrrp4:~# netstat -ntlp
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State                   PID/Program name
tcp        0      0 0.0.0.0:8080             0.0.0.0:*                LISTEN                  988255/nginx: maste
tcp        0      0 0.0.0.0:80               0.0.0.0:*                LISTEN                  988255/nginx: maste
tcp        0      0 0.0.0.0:2320             0.0.0.0:*                LISTEN                  954126/sshd: /usr/s
tcp        0      0 127.0.0.53:53           0.0.0.0:*                LISTEN                  500/systemd-resolve
tcp        0      0 127.0.0.1:631           0.0.0.0:*                LISTEN                  1973523/cupsd
tcp        0      0 0.0.0.0:3000             0.0.0.0:*                LISTEN                  1991099/docker-prox
tcp        0      0 0.0.0.0:443              0.0.0.0:*                LISTEN                  988255/nginx: maste
tcp        0      0 127.0.0.1:33060         0.0.0.0:*                LISTEN                  832110/mysqld
tcp6       0      0 :::3306                 :::*                    LISTEN                  832110/mysqld
tcp6       0      0 :::9100                 :::*                    LISTEN                  1658943/prometheus-
tcp6       0      0 :::80                   :::*                    LISTEN                  988255/nginx: maste
tcp6       0      0 :::2320                 :::*                    LISTEN                  954126/sshd: /usr/s
tcp6       0      0 :::1:631                :::*                    LISTEN                  1973523/cupsd
tcp6       0      0 :::3000                 :::*                    LISTEN                  1991103/docker-prox
tcp6       0      0 :::9113                 :::*                    LISTEN                  1990420/prometheus-
tcp6       0      0 :::443                  :::*                    LISTEN                  988255/nginx: maste
tcp6       0      0 :::9090                 :::*                    LISTEN                  1990902/prometheus
tcp6       0      0 :::9093                 :::*                    LISTEN                  1745713/prometheus-
tcp6       0      0 :::9094                 :::*                    LISTEN                  1745713/prometheus-
```

Открываем веб-морду графаны

admin
admin

<http://37.46.130.3:3000/login>



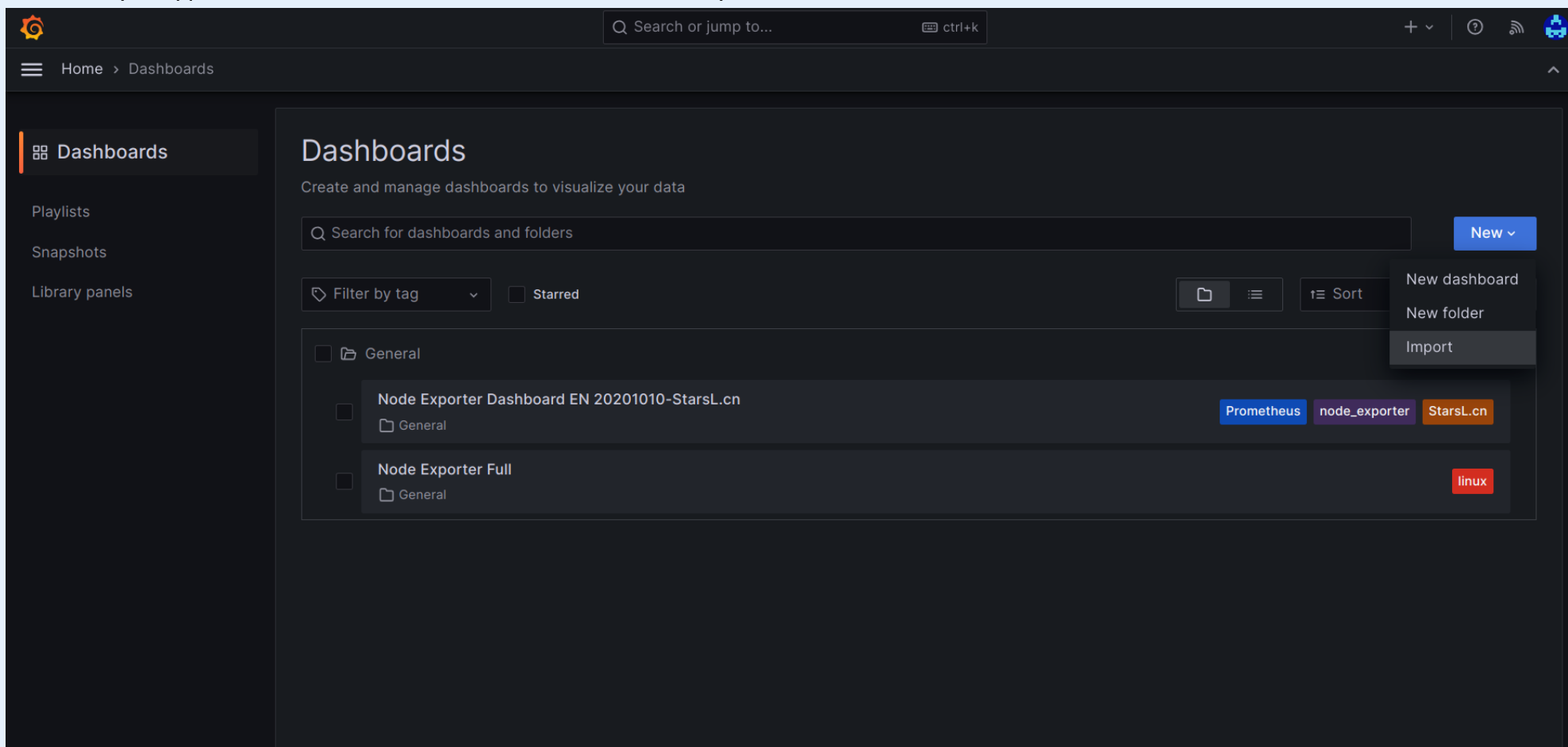
При первой авторизации будет запрошена смена пароля, вводим новый
Затем нужно импортировать дашборд для экспортера (например node-exporter)

Идем на офф. сайт [Grafana](https://grafana.com/) и копируем код нужного дашборда

Например

- [Node Exporter Full](#)
- [Node Exporter Dashboard EN 20201010-StarsL.cn](#)
- [MySQL Exporter Quickstart and Dashboard](#)

Затем переходим в меню Home=>Dashboards и жмем Import




❌ Ошибка связки Grafana & Prometheus

<https://github.com/grafana/grafana/issues/46434>

Post "<http://127.0.0.1:9090/api/v1/query>": dial tcp 127.0.0.1:9090: connect: connection refused - There was an error returned querying the Prometheus API.

если обратиться по пути <http://127.0.0.1:9090/api/v1/query>
то в теле ответа будет

```
{"status": "error", "errorType": "bad_data", "error": "invalid parameter 'query': 1:1: parse error: no expression found in input"}
```



Prometheus

Type: Prometheus

⌵⌵⌵ Settings

⌵⌵⌵ Dashboards

🕒

Alerting supported

Name ⓘ

Prometheus

Default

HTTP

Prometheus server URL ⓘ

http://127.0.0.1:9090/

Allowed cookies ⓘ

New tag (enter key to add)

Add

Timeout ⓘ

Timeout in seconds

Other

Custom query parameters ⓘ

Example: max_source_resolution=5m&timeout

HTTP method ⓘ

POST ⌵

Exemplars

+ Add

❗

Post "http://127.0.0.1:9090/api/v1/query": dial tcp 127.0.0.1:9090: connect: connection refused - There was an error returned querying the Prometheus API.

Delete

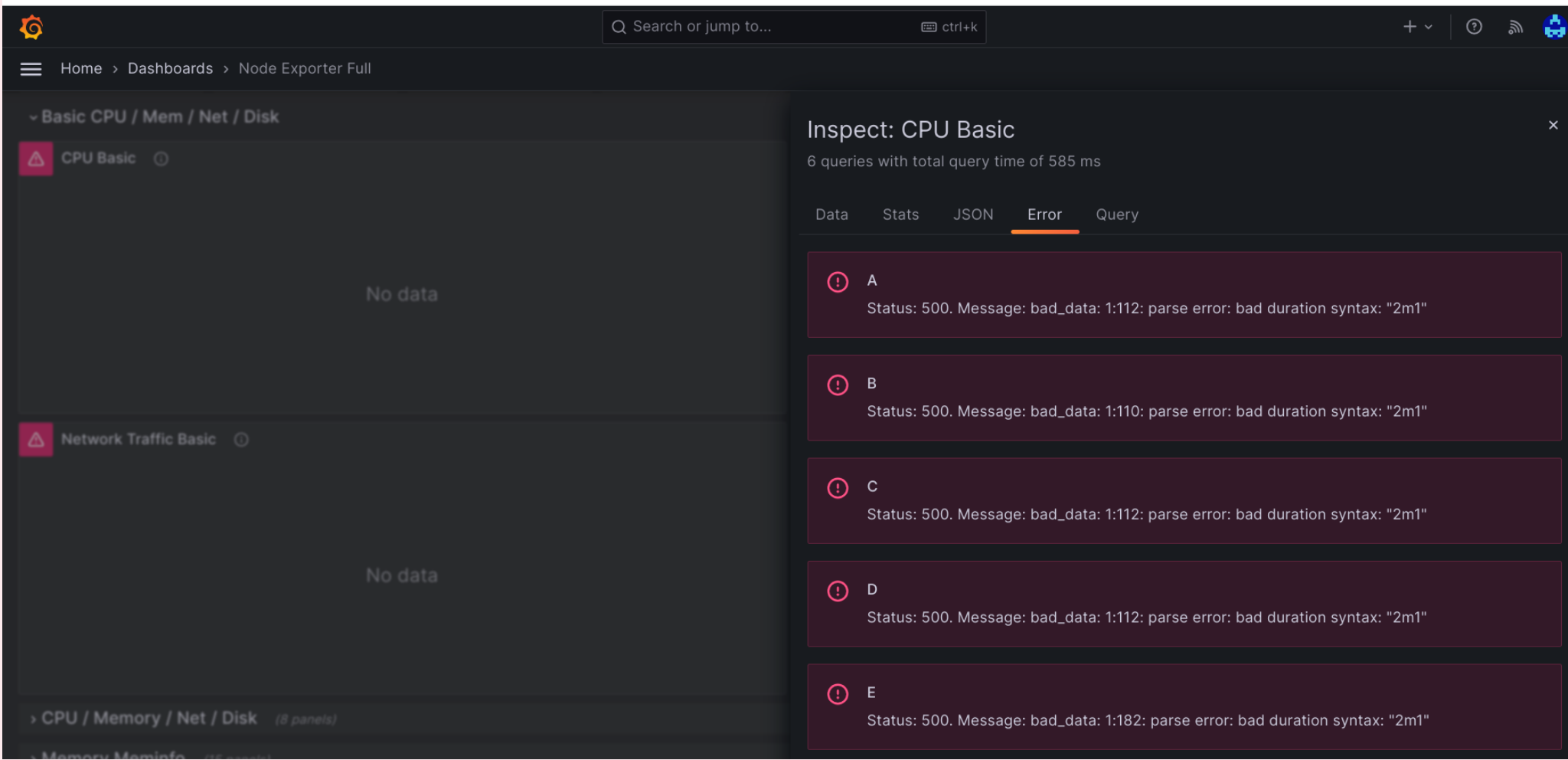
Save & test

В данном случае, причиной ошибки послужило правило в IPTABLES
после его удаления контейнер с Grafana смог достучаться до Prometheus

```
sudo iptables -D INPUT -p tcp -m tcp --dport 9090 -j DROP
```

🚨 Ошибки запросов PromQL из Grafana в Prometheus

после того как дашборд был создан в нем 99% графиков были с ошибками



некоторые из них содержали такой текст ошибки

```
Status: 500. Message: bad_data: invalid parameter 'query': 1:131: parse error: bad duration syntax: "10m1"
```

Как выяснилось причиной являлось то что в запросах использовался новый параметр [\\$__rate_interval](#) вместо [\\$__interval](#) после замены на [\\$__interval](#) все графики начинают работать корректно.