



O.T.U.S
ОНЛАЙН-ОБРАЗОВАНИЕ

Онлайн-образование



Меня хорошо видно && слышно?

Ставьте + , если все хорошо
Напишите в чат, если есть проблемы

НЕ ЗАБЫТЬ ВКЛЮЧИТЬ
ЗАПИСЬ!!!

План

- преимущества и фиши
- установка, конфиг, веб каталог
- меню и дашборды
- плагины
- клауд - централизованный просмотр
- стриминг
- custom dashboard
- импорт и экспорт снэпшотов
- триггеры и алерты
- диапазон хранения
- экспорт по внешние базы - прометей, графит

Преимущества и фичи

- мощные и гибкие дашборды высокой точности
- возможность длительного хранения (до нескольких месяцев)
- полноценный запуск из коробки, преднастроенные метрики без дополнительной конфигурации
- распределенная система без необходимости централизации
- уведомления и алерты
- быстрые легковесные сборщики метрик
- netdata cloud для централизации мониторинга

Установка

```
bash <(curl -Ss https://my-netdata.io/kickstart.sh)
```

- все варианты
- docker
- helm chart для k8s
- rpm, deb

Конфиги и файлы

- порт: 19999
- при установке выбираете где будут находиться файлы
- конфиги
 - /opt/netdata/etc/netdata
 - netdata.conf
- веб приложение
 - /opt/netdata/usr/share/netdata/web/

Возможности дашбордов

- удобная система меню
- масштабируемый таймфрейм (SHIFT или ALT + колесико мышки)
- изменение размера графика (тянуть мышкой за правый нижний угол)
- передвижение по истории графика вправо/влево (с зажатой мышкой)
- сброс в дефолтное состояние (dblclick)
- масштабирование конкретной области выделения (SHIFT + mouse)
- манипулирование отображаемыми измерениями

Аттрибуты дашборда

- dimension
 - метрика отображенная на графике
- family
 - семейство устройств
- context
 - тип метрики

Базовая кастомизация

```
cd /opt/netdata/usr/share/netdata/web/  
cp dashboard_info_custom_example.js otus_dashboard_info.js  
vi /opt/netdata/etc/netdata  
    [web]  
        custom dashboard_info.js = otus_dashboard_info.js
```

```
'system': {  
    title: 'System Overview',  
    icon: '<i class="fas fa-bookmark"></i>',  
    info: 'Overview of the key system metrics.'  
},
```

Создание собственных дашбордов

- документация
- Примеры всех типов графиков
- Пример простого дашборда
- мультиточечные дашборды (при настроенном стриминге),
`dash-example.html`

Плагины

- Internal plugins
- External plugins
- Collector modules (via plugin orchestrators)
 - charts.d.plugin (Bash)
 - go.d.plugin (Go)
 - node.d.plugin (NodeJS)
 - python.d.plugin (Python) Third-party plugins

```
ls -l /opt/netdata/usr/libexec/netdata/plugins.d/  
/opt/netdata/usr/libexec/netdata/plugins.d/go.d.plugin -d -m
```

Демо: включаем плагин для docker

- Модуль для Go
- /etc/docker/daemon.json

```
{  
  "metrics-addr" : "127.0.0.1:9323",  
  "experimental" : true  
}
```

```
sudo -u netdata /usr/libexec/netdata/plugins.d/go.d.plugin -d -m docker  
systemctl restart netdata
```

Alerts

- предустановленные проверки и нотификации
 - `/usr/lib/netdata/conf.d/health.d/`
- дериктория для кастомных алертов
 - `/etc/netdata/health.d`
- скрипт нотификации
 - `/usr/libexec/netdata/plugins.d/alarm-notify.sh`

Пример настройки правила

- описание параметров

```
template: disk_space_usage
  on: disk.space
  os: linux freebsd
  hosts: *
  families: !/dev !/dev/* !/run !/run/* *
  calc: $used * 100 / ($avail + $used)
  units: %
  every: 1m
  warn: $this > (($status >= $WARNING ) ? (80) :
  crit: $this > (($status == $CRITICAL) ? (90) :
  delay: up 1m down 15m multiplier 1.5 max 1h
  info: current disk space usage
  to: sysadmin
```

Кастомизация алертов

- документация
- раздел [health] в netdata.conf
- режим "тишины"
 - `to: silent`
- отключение
 - `to: disabled`
- глобальный конфиг для нотификаций
 - `sudo ./edit-config`
`health_alarm_notify.conf`
- скрипт отправки нотификаций
 - `/usr/libexec/netdata/plugins.d/alarm-notify.sh`

Пример настройки алертинга в телеграм

```
# enable/disable sending telegram messages
SEND_TELEGRAM="YES"

# Contact the bot @BotFather to create a new bot and receive a bot token
# Without it, netdata cannot send telegram messages.
TELEGRAM_BOT_TOKEN="1098563390:AAFho14a-buFqWog7SFd1kenSWXs5_y_u9"

# To get your chat ID send the command /getid to telegram bot @myidbot
# (https://t.me/myidbot). Each user also needs to open a conversation with
# bot that will be sending notifications.
# If a role's recipients are not configured, a message will be sent to
# this chat id (empty = do not send a notification for unconfigured roles)
DEFAULT_RECIPIENT_TELEGRAM="-420083072"
```

Cloud

- клоуд для централизации ссылок на все ваши сервера с нетдатою
- адреса можно добавлять вручную
- либо они появляются сами при посещении
- <https://app.netdata.cloud/>

Streaming

- документация
- использование как headless collector (без внутренней базы и веб интерфейса)
 - <http://parent-host:parent-port/host/child-host/>
- репликация локальной базы
- использование прокси

Опции конфигурации

```
[global]
  memory mode = none | ram | save | map | dbengine

[web]
  mode = none | static-threaded
  accept a streaming request every seconds = 0

[backend]
  enabled = yes | no
  type = graphite | opentsdb
  destination = IP:PORT ...
  update every = 10
```

Опции стриминга

- stream.conf
- sending node

```
[stream]
enabled = yes | no
destination = IP:PORT[:SSL] ...
api key = XXXXXXXXXXXX
```

- receiving node

```
# replace API_KEY with your uuidgen generated
[API_KEY]
enabled = yes
default history = 3600
default memory mode = save
health enabled by default = auto
allow from = *
```

Экспорт метрик

- документация
- graphite, opentsdb, prometheus, TimescaleDB
- prometheus remote write (бинарный сжатый протокол поверх HTTP используемый Elasticsearch, Gnocchi, Graphite, InfluxDB, Kafka, OpenTSDB, PostgreSQL/TimescaleDB, Splunk, VictoriaMetrics)
- ./edit-config exporting.conf

Длительное хранение метрик

- калькулятор

```
[global]
memory mode = dbengine
page cache size = 64
dbengine disk space = 512
```

Использование монгодб

```
[mongodb:my_mongo_instance]
  enabled = yes
  destination = mongodb://localhost
  database = netdata
  collection = netdata_metrics
```

дополнительные возможности дашбордов

- Возможность экспортировать текущее состояние в файл для передачи данных с помощью других каналов связи
- Импорт файла в отдельно стоящую pedata
- панель настроек отображения, производительности и синхронизации


Рефлексия



Отметьте 3 пункта, которые вам запомнились с вебинара



Что вы будете применять в работе из сегодняшнего вебинара?

The image features a blue-tinted aerial view of a dense city skyline, likely New York City, with numerous skyscrapers. A semi-transparent blue overlay with a white network pattern of lines and dots is positioned in the center, containing the text. The text is white and reads: "Заполните, пожалуйста, опрос о занятии по ссылке в чате".

Заполните, пожалуйста,
опрос о занятии по ссылке в чате