**Aplana performance testing framework (Ptf). Платформа для проведения нагрузочного тестирования**

***Руководство по установке***

**Дата:** 04.02.2019

Содержание

[1. Введение 3](#_Toc180157)

[2. Установка DOCKER и DOCKER-COMPOSE 5](#_Toc180158)

[3. Инициализация и подключение к SWARM 6](#_Toc180159)

[4. Создание и запуск Docker-контейнеров influxDB, grafana, telegraf, jmeterbase, jenkins, redis 7](#_Toc180160)

[5. Запуск и остановка контейнеров 11](#_Toc180161)

[6. Создание и запуск контейнера redis, подключение к OVERLAY-сети 12](#_Toc180162)

[7. Создание overlay-сети 13](#_Toc180163)

[8. Шаблоны Grafana 14](#_Toc180164)

[9. Настройка JenkinS 17](#_Toc180165)

[10. ПРИЛОЖЕНИЕ А. ПРИМЕР ШАБЛОНА ДЛЯ GRAFANA 21](#_Toc180166)

# Введение

* 1. **Назначение документа**

Данное руководство предназначено для администраторов, осуществляющих настройку платформы для проведения нагрузочного тестирования. В руководстве описываются процедуры подготовки ведущего сервера к работе.

* 1. **Термины, определения и сокращения**

|  |  |
| --- | --- |
| **Название** | **Определение** |
| **Система** | Программный комплекс Aplana PTF |
| **ПО** | Программное обеспечение |
| **НТ** | Нагрузочное тестирование |
| **БД** | База данных |
| **СУБД** | Система управления базами данных |
| **ОС** | Операционная система |

* 1. **Системные требования**

Серверная часть (сервер приложений и баз данных):

* Процессор – не менее 2vCPU, 3 GHz+
* Оперативная память – не менее 16 GB
* Дисковая память – объем не менее 50 GB для Системы, не менее 150 GB для БД.
* Пропускная способность внешнего канала связи – не менее 100 Мбит/сек 2)

Клиентская часть гарантированно работает в web-браузерах:

* Google Chrome
* Internet Explorer 10+.
  1. **Требования к программному обеспечению**

На сервере приложений и БД должно быть предустановлено следующее ПО:

1. ОС GNU/Linux либо Unix
2. Выделены права sudo
   1. **Требования к квалификации персонала**

Пользователь, выполняющий операции по установке, обновлению и переносу Системы должен обладать знаниями ОС Linux на уровне не ниже администратора.

* 1. **Порядок действий развертывания Aplana PTF**

**Для master-сервера:**

1. Установить docker и docker-compose согласно инструкции по установке;
2. Инициализировать swarm;
3. Создать и запустить контейнеры influxDB, grafana, telegraf, jmeterbase, jenkins, redis при помощи docker-compose;
4. Создать overlay сеть.

**Для slave-сервера:**

1. Установить docker и docker-compose согласно инструкции по установке;
2. Подключиться к swarm;
3. Создать и запустить контейнеры telegraf, jmeterbase, jenkins при помощи docker-compose;
4. Запустить контейнер redis (и подсоединиться к overlay-сети на master-сервере).

# Установка DOCKER и DOCKER-COMPOSE

Требования:

* ОС GNU/Linux либо Unix
* Права sudo

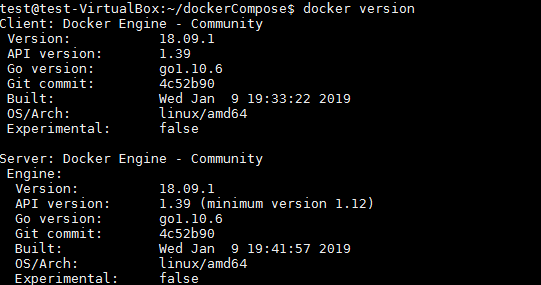
1. Скопировать архив install.tar в $HOME – домашнюю директорию пользователя. Выполнить следующие команды:

tar xvf $HOME/install.tar

sudo sh $HOME/install.sh > install.log

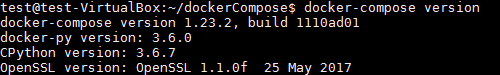
1. Проверить статус установки:

docker -version



1. Убедиться, что установка docker-compose прошла успешно:

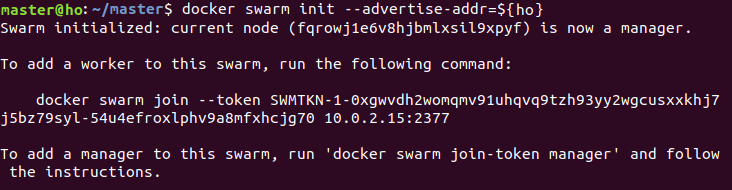
docker-compose --version



# Инициализация и подключение к SWARM

**Инициализация swarm:**

Выполнить команду:

 docker swarm init --advertise-addr=${hostname}

Получим token, который затем будем использовать при подключении slave-сервера.

**Подключение к swarm:**

Выполнить полученную на master-сервере команду (в данном случае такую:)

docker swarm join --token SWMTKN-1-0xgwvdh2womqmv91uhqvq9tzh93yy2wgcusxxkhj7j5bz79syl-54u4efroxlphv9a8mfxhcjg70 10.0.2.15:2377

# Создание и запуск Docker-контейнеров influxDB, grafana, telegraf, jmeterbase, jenkins, redis

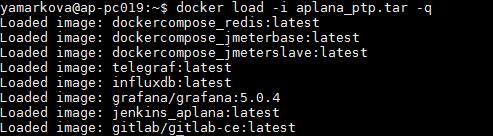
1. Скопировать файлы aplana\_ptp.tar и volumes.tar в домашнюю директорию $HOME
2. Установить образы контейнеров из архива
3. Выполнить команду sudo docker-compose up

tar xvf $HOME/volumes.tar

docker load –i aplana\_ptp.tar –q

cd dockerCompose

docker-compose up



В файле docker-compose.yml для master содержится следующая информация:

version: '3.1'

services:

jmeterbase:

build:

context: ./

dockerfile: "./jmeter/jmeterbase"

container\_name: jmeterbase

tty: true

volumes:

- "${HOME}/scripts:/external"

ports:

- "55555:55555"

links:

- influxdb

networks:

- aplana

jmeterslave:

build:

context: ./

dockerfile: "./jmeter/jmeterbase"

container\_name: slave

tty: true

volumes:

- "${HOME}/scripts:/external"

ports:

- "50002:50002"

- "10099:10099"

links:

- influxdb

entrypoint: /jmeter/apache-jmeter-5.0/bin/jmeter-server -Dserver.rmi.localport=50001 -Dserver\_port=10099 -Dserver.rmi.ssl.disable=true

networks:

- aplana

influxdb:

image: influxdb:latest

container\_name: influxdb

restart: always

environment:

INFLUXDB\_ADMIN\_ENABLED: "true"

INFLUXDB\_DB: jmeter

INFLUXDB\_ADMIN\_USER: admin

INFLUXDB\_ADMIN\_PASSWORD: admin

volumes:

- "influxdb\_volume:/var/lib/influxdb"

ports:

- "8083:8083"

- "8086:8086"

- "8090:8090"

networks:

- aplana

telegraf:

image: telegraf:latest

container\_name: telegraf

links:

- influxdb

volumes:

- "./telegraf/telegraf.conf:/etc/telegraf/telegraf.conf:ro"

- "/var/run/docker.sock:/var/run/docker.sock"

networks:

- aplana

grafana:

image: grafana/grafana:5.0.4

container\_name: grafana

restart: always

env\_file: ./grafana/configuration.env

volumes:

- "grafana\_volume:/var/lib/grafana"

- "./grafana/provisioning/:/etc/grafana/provisioning/"

- "./grafana/templates/:/var/lib/grafana/dashboards/"

ports:

- "3000:3000"

links:

- influxdb

networks:

- aplana

jenkins:

image: 'jenkins\_aplana:latest'

container\_name: jenkins

restart: always

volumes:

- "${HOME}/jenkins/jobs:/var/jenkins\_home/jobs"

- "/var/run/docker.sock:/var/run/docker.sock"

- "${HOME}/report:/report"

- "${HOME}/jenkins/userContent:/var/jenkins\_home/userContent"

- "${HOME}/jenkins/icons:/var/jenkins\_home/userContent/icons"

- "${HOME}/jenkins/io.jenkins.plugins.logintheme.LoginTheme.xml:/var/jenkins\_home/io.jenkins.plugins.logintheme.LoginTheme.xml"

- "${HOME}/jenkins/org.codefirst.SimpleThemeDecorator.xml:/var/jenkins\_home/org.codefirst.SimpleThemeDecorator.xml"

ports:

- "8080:8080"

- "50000:50000"

networks:

- aplana

redis:

build: redis

container\_name: redismaster

ports:

- "6379:6379"

networks:

- aplana

gitlab:

image: 'gitlab/gitlab-ce:latest'

container\_name: gitlab

restart: always

hostname: 'gitlab.example.com'

environment:

GITLAB\_OMNIBUS\_CONFIG: |

external\_url 'http://gitlab.example.com:9090'

gitlab\_rails['gitlab\_shell\_ssh\_port'] = 2224

ports:

- '9090:9090'

- '2224:22'

volumes:

- '/srv/gitlab/config:/etc/gitlab'

- '/srv/gitlab/logs:/var/log/gitlab'

- '/srv/gitlab/data:/var/opt/gitlab'

volumes:

influxdb\_volume:

grafana\_volume:

jenkins\_volume:

networks:

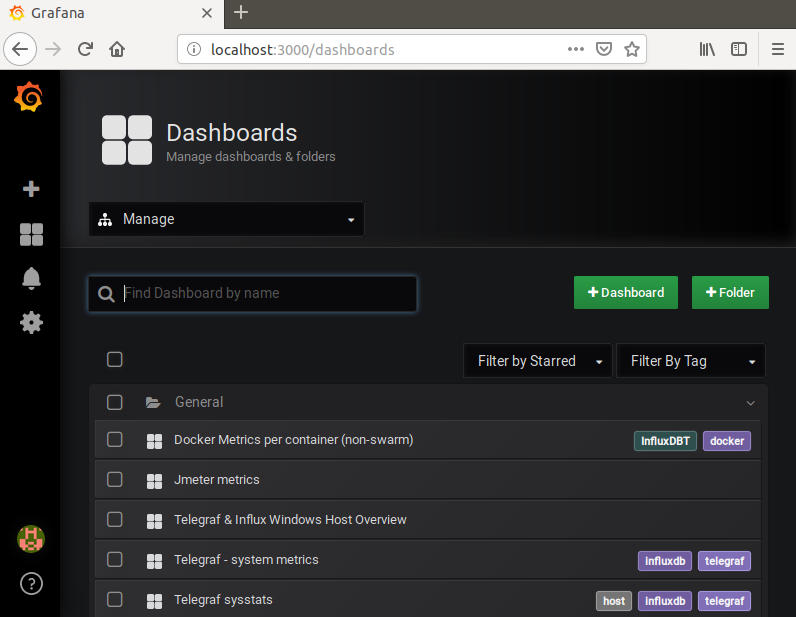
aplana:

driver: bridge

1. Можно проверить, что появились тома данных (volume):

docker volume ls

1. А также (для мастера) присутствие шаблонов в Grafana на http://{ip-address}:3000/dashboards/, логин: admin, пароль: admin.



# Запуск и остановка контейнеров

Запускать или останавливать контейнеры нужно следующими командами:

docker start influxdb

docker start grafana

docker start telegraf

docker start jmeterbase

docker start redismaster

docker start jenkins

docker stop influxdb

docker stop grafana

docker stop telegraf

docker stop jmeterbase

docker stop redismaster

docker stop jenkins

# Создание и запуск контейнера redis, подключение к OVERLAY-сети

Запуск образа redis-slave

docker run -it -v /home/aplana/redis/redis.conf:/usr/local/etc/redis/redis.conf --name redis-slave --net ptp-net redis redis-server --slaveof redis-master 6379

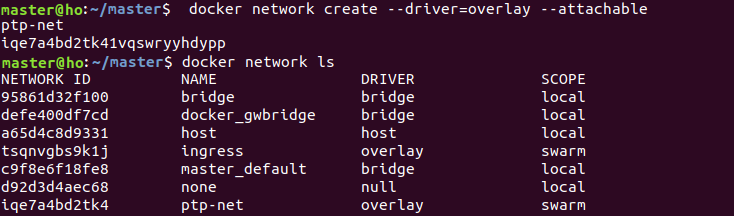
# Создание overlay-сети

Для создания overlay сети на мастере выполнить команду:

docker network create --driver=overlay --attachable ptp-net

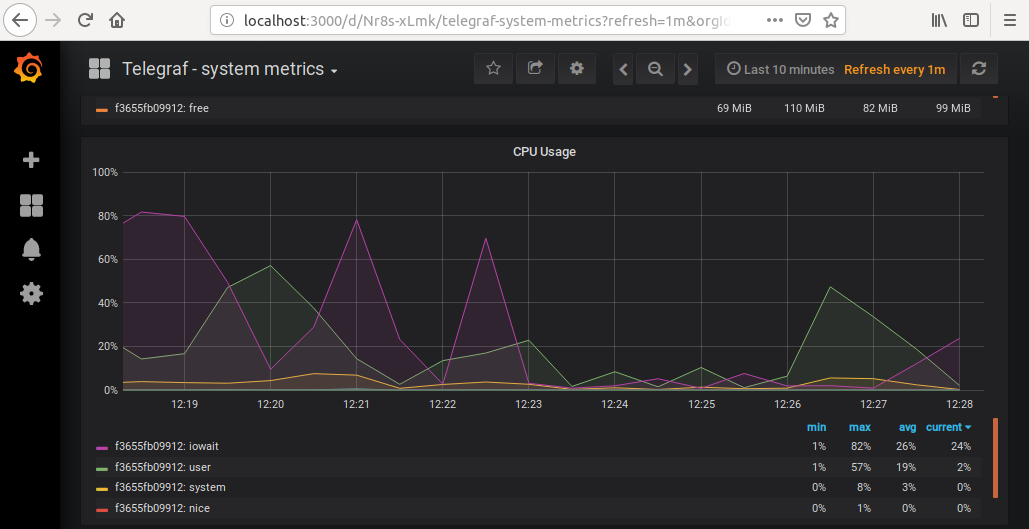
Можно проверить, что сеть с названием ptp-net создана:

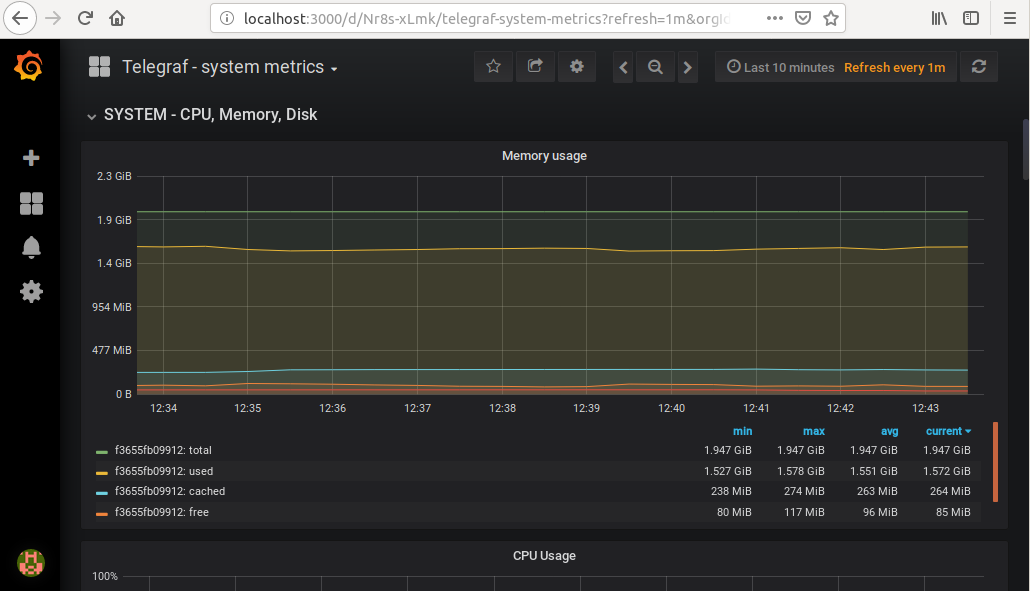
docker network ls



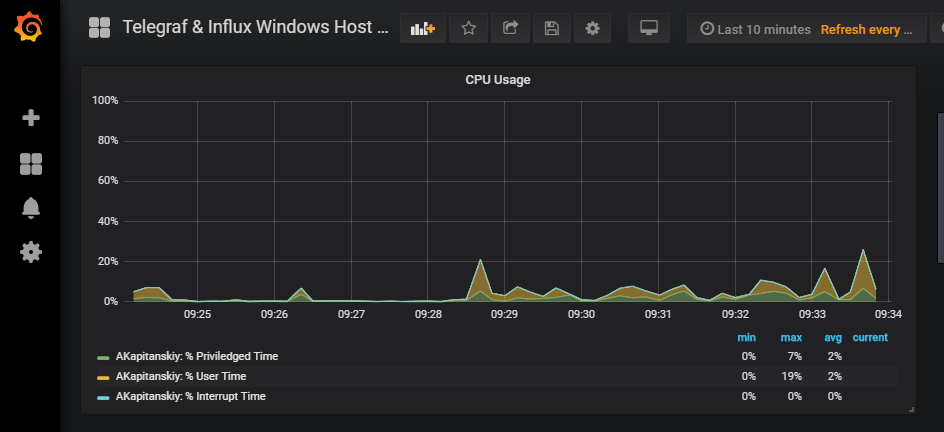
# Шаблоны Grafana

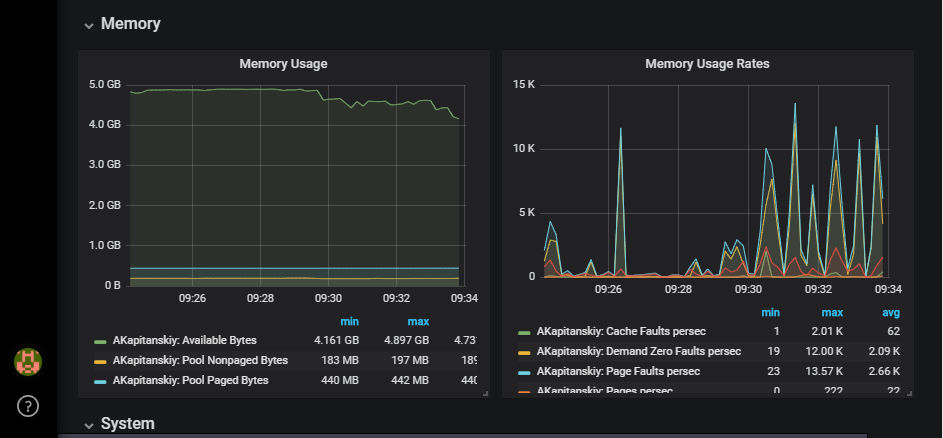
В Grafana по ссылке http://localhost:3000/d/Nr8s-xLmk/telegraf-system-metrics?refresh=1m&orgId=1 отображаются системные метрики для Linux. Примеры:



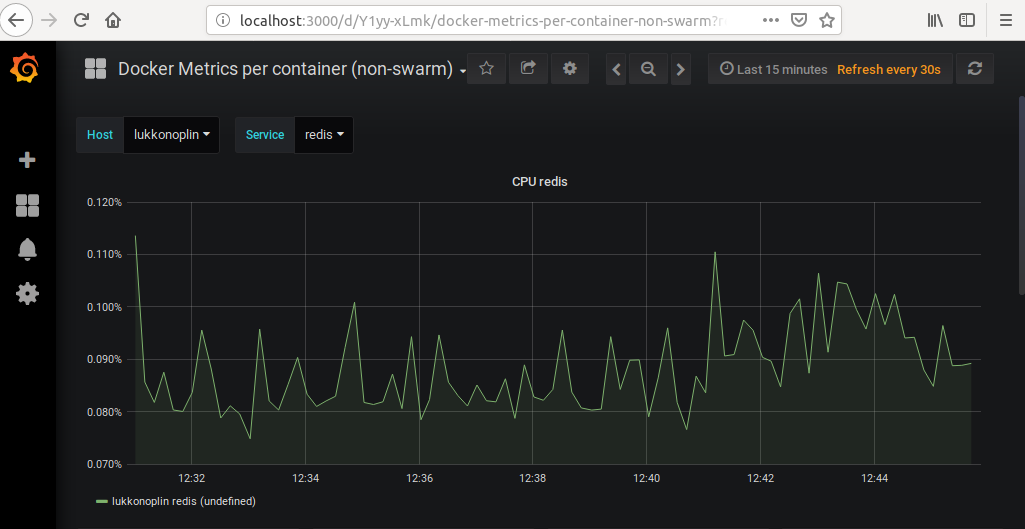
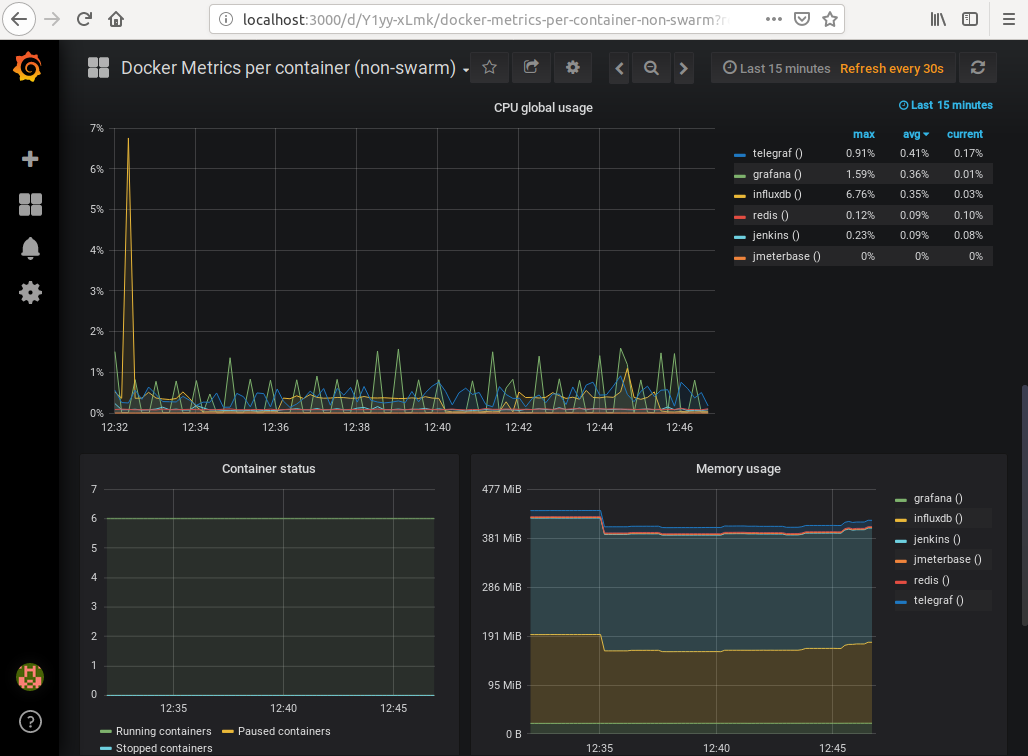


По ссылке <http://localhost:3000/d/ps2nh4pmk/telegraf-and-influx-windows-host-overview1?refresh=1m&orgid=1&from=now-10m&to=now> доступны системные метрики для Windows:





На <http://localhost:3000/d/Y1yy-xLmk/docker-metrics-per-container-non-swarm?refresh=30s&orgId=1> отображаются метрики внутри контейнеров. Есть возможность выбрать контейнеры, информация по которым интересует.

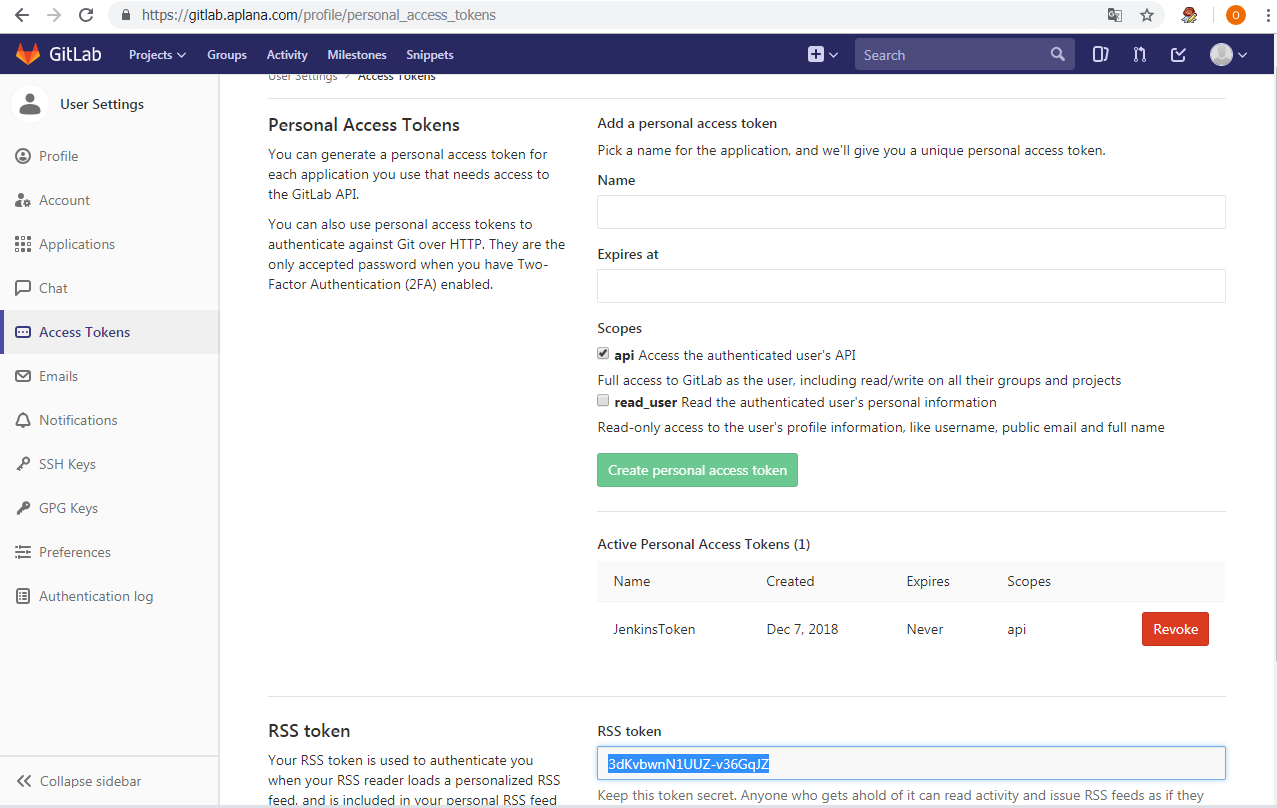


# Настройка JenkinS

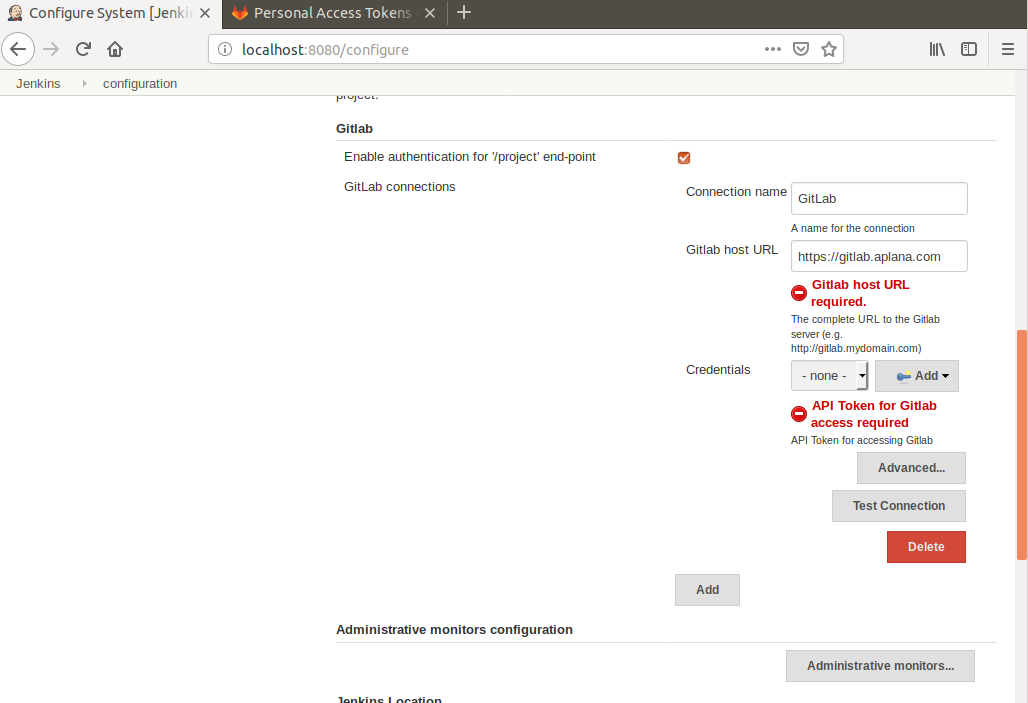
Для Jenkins все необходимые плагины были загружены при создании контейнера.

**Подключение к GitLab**

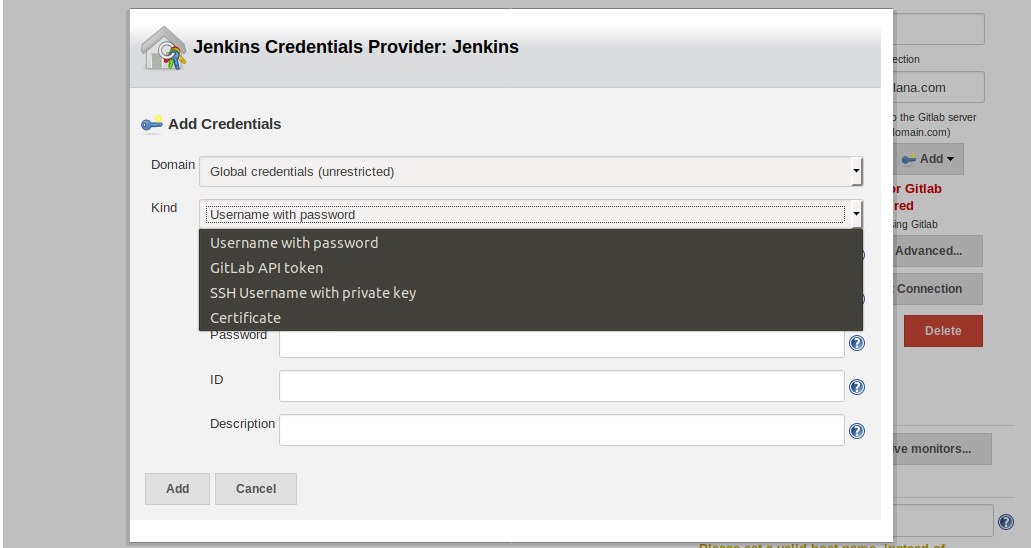
1. На GitLab в настройках профиля выбрать раздел Access Tokens.
2. Поставить галочку на **api** Access the authenticated user's API и нажать кнопку Create personal access token. Ниже под строкой RSS token появляется токен, который будет далее использоваться.

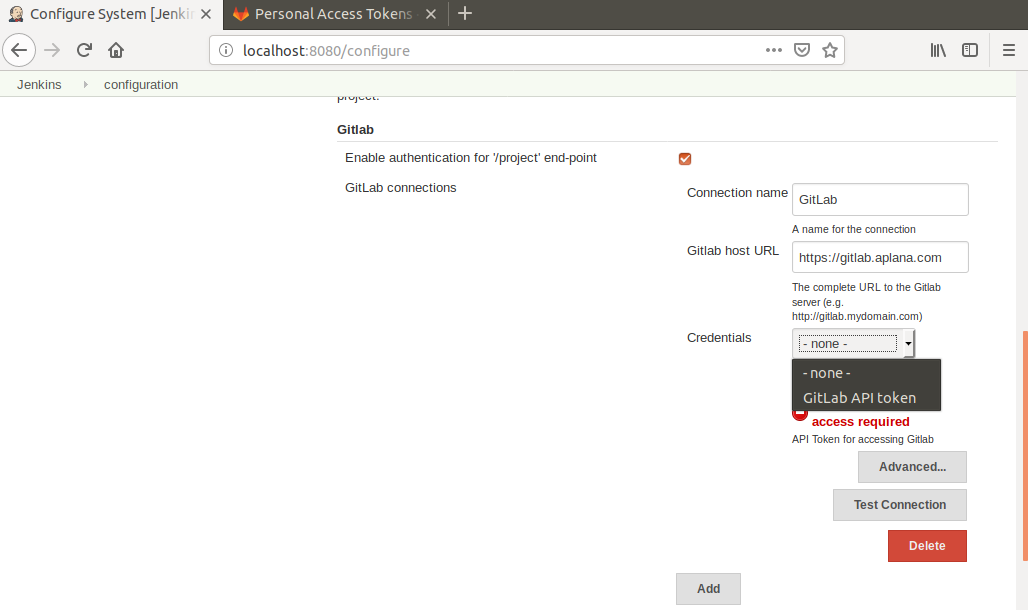


1. В разделе Manage Jenkins -> Configure System пролистать до настроек GitLab.
2. Ввести название подключения и адрес GitLab.
3. В Credentials нажать Add и в выпадающем меню выбрать Jenkins.



1. В появившемся окне во вкладке Kind выбрать GitLab API token
2. В качестве API token ввести тот, который был получен на GitLab, нажать кнопку Add.



1. Теперь в Credentials выбрать в выпадающем меню GitLab API token и нажать кнопку Save.

**Подключение slave-сервера**

На master-сервере выполнить действия:

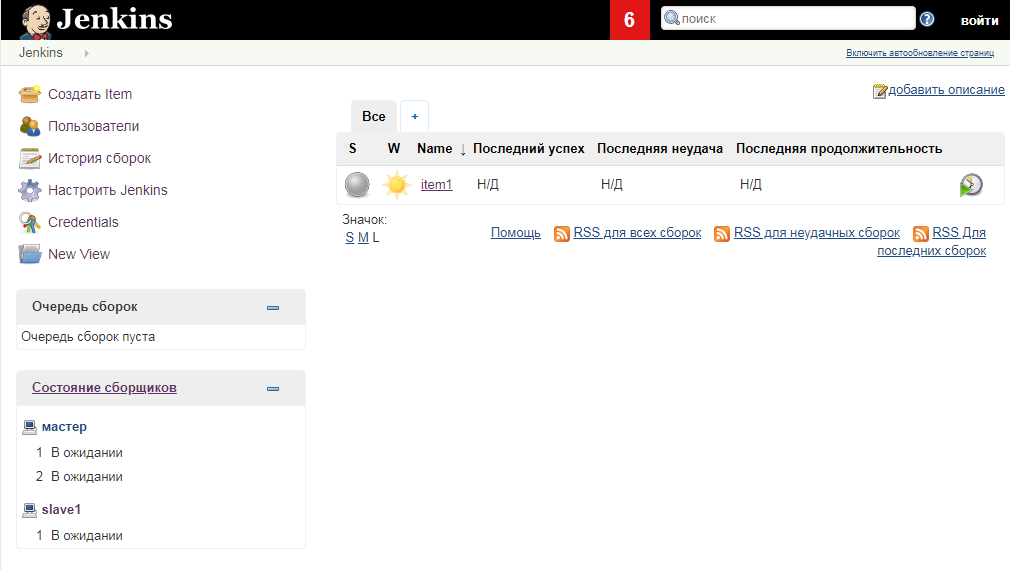
1. В разделе Manage Jenkins -> Управление средами сборки нажать на кнопку Новый узел.
2. Ввести название узла. Для первого создаваемого узла выбрать Permanent Agent, нажать ОК.
3. Задать настройки создаваемого узла. Обязательно заполнить поле Корень удаленной ФС (например, /home/jenkins). В качестве Способа запуска выбрать Launch agent via Java Web Start. В конце нажать Save.

На slave-сервере выполнить:

1. Зайти на страницу http://yourjenkinsmasterhost:8080/computer/, нажать на созданный узел.
2. Скачать agent.jar, нажав на него.
3. В директории с agent.jar выполнить команду, которая была приведена на странице, откуда скачивался agent.jar

java -jar [agent.jar](http://192.168.100.150:8080/jnlpJars/agent.jar) -jnlpUrl http:// yourjenkinsmasterhost:8080/computer/your\_slavenode\_name/slave-agent.jnlp -workDir "your\_directory"

1. Теперь при создании задачи в настройках можно поставить галочку на Это – параметризованная сборка и в параметрах выбрать node, и серверы(сборщики) и их состояния будут отображаться слева, а также появилась возможность выбора, на каком из них выполнять задачу.



# ПРИЛОЖЕНИЕ А. ПРИМЕР ШАБЛОНА ДЛЯ GRAFANA

{

"\_\_inputs": [

{

"name": "DS\_TEST",

"label": "test",

"description": "",

"type": "datasource",

"pluginId": "influxdb",

"pluginName": "InfluxDB"

}

],

"\_\_requires": [

{

"type": "grafana",

"id": "grafana",

"name": "Grafana",

"version": "5.0.4"

},

{

"type": "panel",

"id": "graph",

"name": "Graph",

"version": "5.0.0"

},

{

"type": "datasource",

"id": "influxdb",

"name": "InfluxDB",

"version": "5.0.0"

}

],

"annotations": {

"list": [

{

"builtIn": 1,

"datasource": "-- Grafana --",

"enable": true,

"hide": true,

"iconColor": "rgba(0, 211, 255, 1)",

"limit": 100,

"name": "Annotations & Alerts",

"showIn": 0,

"type": "dashboard"

},

{

"datasource": "${DS\_TEST}",

"enable": true,

"hide": false,

"iconColor": "rgba(255, 96, 96, 1)",

"limit": 100,

"name": "Начало теста",

"query": "SELECT \* from events WHERE $timeFilter and text=~ /.\*started.\*/",

"showIn": 0,

"tags": [],

"tagsColumn": "text",

"textColumn": "text",

"type": "tags"

},

{

"datasource": "${DS\_TEST}",

"enable": true,

"hide": false,

"iconColor": "rgba(255, 96, 96, 1)",

"limit": 100,

"name": "Конец теста",

"query": "SELECT \* from events WHERE $timeFilter and text=~ /.\*ended.\*/",

"showIn": 0,

"tags": [],

"tagsColumn": "text",

"textColumn": "text",

"type": "tags"

}

]

},

"editable": true,

"gnetId": null,

"graphTooltip": 0,

"id": null,

"iteration": 1537273823645,

"links": [],

"panels": [

{

"aliasColors": {},

"bars": false,

"dashLength": 10,

"dashes": false,

"datasource": "${DS\_TEST}",

"fill": 0,

"gridPos": {

"h": 9,

"w": 12,

"x": 0,

"y": 0

},

"id": 4,

"legend": {

"alignAsTable": false,

"avg": false,

"current": true,

"max": false,

"min": false,

"rightSide": false,

"show": true,

"total": false,

"values": true

},

"lines": true,

"linewidth": 2,

"links": [],

"nullPointMode": "null",

"percentage": false,

"pointradius": 1,

"points": true,

"renderer": "flot",

"seriesOverrides": [],

"spaceLength": 10,

"stack": false,

"steppedLine": false,

"targets": [

{

"alias": "Кол-во виртуальных пользователей",

"groupBy": [

{

"params": [

"$aggregation"

],

"type": "time"

},

{

"params": [

"0"

],

"type": "fill"

}

],

"measurement": "jmeter",

"orderByTime": "ASC",

"policy": "default",

"refId": "B",

"resultFormat": "time\_series",

"select": [

[

{

"params": [

"meanAT"

],

"type": "field"

},

{

"params": [],

"type": "mean"

}

]

],

"tags": []

}

],

"thresholds": [],

"timeFrom": null,

"timeShift": null,

"title": "Кол-во виртуальных пользователей",

"tooltip": {

"shared": true,

"sort": 0,

"value\_type": "individual"

},

"type": "graph",

"xaxis": {

"buckets": null,

"mode": "time",

"name": null,

"show": true,

"values": []

},

"yaxes": [

{

"format": "short",

"label": null,

"logBase": 1,

"max": null,

"min": null,

"show": true

},

{

"format": "short",

"label": null,

"logBase": 1,

"max": null,

"min": null,

"show": true

}

]

},

{

"aliasColors": {},

"bars": true,

"dashLength": 10,

"dashes": false,

"datasource": "${DS\_TEST}",

"fill": 1,

"gridPos": {

"h": 9,

"w": 12,

"x": 12,

"y": 0

},

"id": 7,

"legend": {

"alignAsTable": true,

"avg": false,

"current": true,

"max": false,

"min": false,

"rightSide": true,

"show": true,

"sideWidth": 250,

"total": false,

"values": true

},

"lines": true,

"linewidth": 1,

"links": [],

"nullPointMode": "null as zero",

"percentage": false,

"pointradius": 5,

"points": false,

"renderer": "flot",

"seriesOverrides": [],

"spaceLength": 10,

"stack": false,

"steppedLine": false,

"targets": [

{

"alias": "$tag\_transaction",

"groupBy": [

{

"params": [

"$aggregation"

],

"type": "time"

},

{

"params": [

"transaction"

],

"type": "tag"

},

{

"params": [

"0"

],

"type": "fill"

}

],

"measurement": "jmeter",

"orderByTime": "ASC",

"policy": "default",

"query": "SELECT mean(\"count\") FROM \"jmeter\" WHERE $timeFilter GROUP BY time(1s),\"transaction\" fill(null)",

"rawQuery": false,

"refId": "A",

"resultFormat": "time\_series",

"select": [

[

{

"params": [

"countError"

],

"type": "field"

},

{

"params": [],

"type": "mean"

}

]

],

"tags": [

{

"key": "transaction",

"operator": "=~",

"value": "/^$transaction$/"

}

]

}

],

"thresholds": [],

"timeFrom": null,

"timeShift": null,

"title": "Кол-во ошибок",

"tooltip": {

"shared": true,

"sort": 0,

"value\_type": "individual"

},

"transparent": false,

"type": "graph",

"xaxis": {

"buckets": null,

"mode": "time",

"name": null,

"show": true,

"values": []

},

"yaxes": [

{

"format": "ops",

"label": null,

"logBase": 1,

"max": null,

"min": null,

"show": true

},

{

"format": "short",

"label": null,

"logBase": 1,

"max": null,

"min": null,

"show": true

}

]

},

{

"aliasColors": {},

"bars": true,

"dashLength": 10,

"dashes": false,

"datasource": "${DS\_TEST}",

"fill": 1,

"gridPos": {

"h": 7,

"w": 24,

"x": 0,

"y": 9

},

"id": 6,

"legend": {

"alignAsTable": true,

"avg": false,

"current": true,

"max": false,

"min": false,

"rightSide": true,

"show": true,

"sideWidth": 250,

"total": false,

"values": true

},

"lines": true,

"linewidth": 1,

"links": [],

"nullPointMode": "null as zero",

"percentage": false,

"pointradius": 5,

"points": false,

"renderer": "flot",

"seriesOverrides": [],

"spaceLength": 10,

"stack": false,

"steppedLine": false,

"targets": [

{

"alias": "$tag\_transaction",

"groupBy": [

{

"params": [

"$aggregation"

],

"type": "time"

},

{

"params": [

"transaction"

],

"type": "tag"

},

{

"params": [

"null"

],

"type": "fill"

}

],

"measurement": "jmeter",

"orderByTime": "ASC",

"policy": "default",

"query": "SELECT mean(\"count\") FROM \"jmeter\" WHERE $timeFilter GROUP BY time(1s),\"transaction\" fill(null)",

"rawQuery": false,

"refId": "A",

"resultFormat": "time\_series",

"select": [

[

{

"params": [

"count"

],

"type": "field"

},

{

"params": [],

"type": "mean"

}

]

],

"tags": [

{

"key": "transaction",

"operator": "=~",

"value": "/^$transaction$/"

}

]

}

],

"thresholds": [],

"timeFrom": null,

"timeShift": null,

"title": "Интенсивность, оп/сек",

"tooltip": {

"shared": true,

"sort": 0,

"value\_type": "individual"

},

"transparent": false,

"type": "graph",

"xaxis": {

"buckets": null,

"mode": "time",

"name": null,

"show": true,

"values": []

},

"yaxes": [

{

"format": "ops",

"label": null,

"logBase": 1,

"max": null,

"min": null,

"show": true

},

{

"format": "short",

"label": null,

"logBase": 1,

"max": null,

"min": null,

"show": true

}

]

},

{

"aliasColors": {

"all": "#99440a"

},

"bars": false,

"dashLength": 10,

"dashes": false,

"datasource": "${DS\_TEST}",

"fill": 0,

"gridPos": {

"h": 9,

"w": 24,

"x": 0,

"y": 16

},

"id": 2,

"legend": {

"alignAsTable": true,

"avg": true,

"current": false,

"hideEmpty": false,

"hideZero": false,

"max": true,

"min": false,

"rightSide": true,

"show": true,

"sideWidth": 250,

"total": false,

"values": true

},

"lines": true,

"linewidth": 2,

"links": [],

"nullPointMode": "connected",

"percentage": false,

"pointradius": 1,

"points": true,

"renderer": "flot",

"seriesOverrides": [],

"spaceLength": 10,

"stack": false,

"steppedLine": false,

"targets": [

{

"alias": "$tag\_transaction",

"groupBy": [

{

"params": [

"$aggregation"

],

"type": "time"

},

{

"params": [

"transaction"

],

"type": "tag"

},

{

"params": [

"0"

],

"type": "fill"

}

],

"measurement": "jmeter",

"orderByTime": "ASC",

"policy": "default",

"refId": "A",

"resultFormat": "time\_series",

"select": [

[

{

"params": [

"pct90.0"

],

"type": "field"

},

{

"params": [],

"type": "mean"

}

]

],

"tags": [

{

"key": "transaction",

"operator": "=~",

"value": "/^$transaction$/"

}

]

}

],

"thresholds": [],

"timeFrom": null,

"timeShift": null,

"title": "Время отклика бизнес-операций",

"tooltip": {

"shared": true,

"sort": 0,

"value\_type": "individual"

},

"type": "graph",

"xaxis": {

"buckets": null,

"mode": "time",

"name": null,

"show": true,

"values": []

},

"yaxes": [

{

"format": "ms",

"label": null,

"logBase": 1,

"max": null,

"min": null,

"show": true

},

{

"format": "short",

"label": null,

"logBase": 1,

"max": null,

"min": null,

"show": true

}

]

}

],

"refresh": "5s",

"schemaVersion": 16,

"style": "dark",

"tags": [],

"templating": {

"list": [

{

"allValue": null,

"current": {

"text": "5s",

"value": "5s"

},

"hide": 0,

"includeAll": false,

"label": "Интервал аггрегации",

"multi": false,

"name": "aggregation",

"options": [

{

"selected": false,

"text": "1s",

"value": "1s"

},

{

"selected": true,

"text": "5s",

"value": "5s"

},

{

"selected": false,

"text": "10s",

"value": "10s"

},

{

"selected": false,

"text": "30s",

"value": "30s"

},

{

"selected": false,

"text": "1m",

"value": "1m"

},

{

"selected": false,

"text": "5m",

"value": "5m"

},

{

"selected": false,

"text": "15m",

"value": "15m"

}

],

"query": "1s,5s,10s,30s,1m,5m,15m",

"type": "custom"

},

{

"allValue": null,

"current": {},

"datasource": "${DS\_TEST}",

"hide": 0,

"includeAll": false,

"label": "Наименование операции",

"multi": true,

"name": "transaction",

"options": [],

"query": "SHOW TAG VALUES FROM \"jmeter\" WITH KEY = \"transaction\"",

"refresh": 1,

"regex": "",

"sort": 0,

"tagValuesQuery": "",

"tags": [],

"tagsQuery": "",

"type": "query",

"useTags": false

}

]

},

"time": {

"from": "now-15m",

"to": "now"

},

"timepicker": {

"refresh\_intervals": [

"5s",

"10s",

"30s",

"1m",

"5m",

"15m",

"30m",

"1h",

"2h",

"1d"

],

"time\_options": [

"5m",

"15m",

"1h",

"6h",

"12h",

"24h",

"2d",

"7d",

"30d"

]

},

"timezone": "",

"title": "Шаблон",

"uid": "iJ8\_t8Tmk",

"version": 2

}